



FACULTAD DE INFORMÁTICA

TESINA DE LICENCIATURA

Título: Juego Educativo Móvil Colaborativo

Autores: Apezteguía Matías, Rapetti Darío Ezequiel

Director: Dra. Gordillo Silvia

Codirector: Dra. Challiol Cecilia

Asesor profesional:

Carrera: Licenciatura en Sistemas

Resumen

A partir del avance tecnológico en estos últimos años, es constante la aparición de nuevos y poderosos dispositivos móviles que ponen esta tecnología al alcance de todos.

Utilizando estos medios como herramientas educativas, se puede acceder a un método de enseñanza en el que los alumnos aprendan jugando.

Con el uso de dispositivos móviles, se pueden generar desarrollos que fomenten un aprendizaje fuera del aula, donde los alumnos o participantes adquieran conocimientos a partir de información relativa a la ubicación donde se encuentren.

Teniendo esto en cuenta y agregando la posibilidad de generar aprendizaje a través de juegos educativos, y además la posibilidad de interactuar y colaborar entre los diferentes participantes involucrados fomentando un aprendizaje colaborativo, obtenemos las bases para desarrollar una aplicación educativa móvil y colaborativa.

El objetivo entonces es, a partir de una investigación de los aspectos inherentes, proponer un modelo general para juegos colaborativos móviles y, a partir de este, realizar un prototipo funcional de un juego educativo, móvil y colaborativo.

Palabras Claves

Juegos Educativos Móvil Colaborativo. Aprendizaje Móvil. Aprendizaje Colaborativo. Juegos Móviles. Juegos basados en Posicionamiento. Computación Móvil.

Trabajos Realizados

Se investigaron y analizaron aspectos relacionados con la computación móvil aplicada a la educación, haciendo hincapié en el aprendizaje colaborativo aprovechando el papel que juega la información espacial, es decir, la información asociada a una ubicación específica.

A partir de dicha investigación, se obtuvo un modelo general para Juegos Colaborativos Móviles, el cual se usó de base para implementar un prototipo funcional de un juego colaborativo móvil (aplicaciones Android que se conectan y comunican con un Servidor Web). Además, se implementó una interfaz de configuración Web para la creación de este tipo de juegos.

Conclusiones

Propusimos un modelo de Juego Colaborativo Móvil que usamos como base para implementar un prototipo funcional. El cual se convierte en un soporte para el aprendizaje colaborativo fuera del aula, buscando que sea fácilmente configurable, dinámico y sencillo de comprender para los participantes involucrados.

Hemos logrado introducirnos en aspectos relativos al aprendizaje colaborativo y móvil, incorporando conocimientos que nos permitirán involucrarnos en futuros desarrollos en temáticas vinculadas.

Trabajos Futuros

- Mejorar el método de comunicación entre las aplicaciones nativas y el servidor web.
- Recuperación ante caídas o fallos del sistema.
- Mejorar manejo de excepciones
- Evaluar métodos de acceso a los puntos de interés.
- Aspectos de diseño del prototipo, alertas, vibraciones en dispositivos móviles, etc.
- Agregar posibilidad de tomar fotografías y/o videos como parte del juego.
- Realizar trabajos de campo.

Índice

1. Introducción	3
1.1. Motivación	3
1.2. Objetivos	4
1.3. Organización de la tesina	4
2. Background	6
2.1. Computación móvil	6
2.2. Contexto en aplicaciones móviles	9
2.2.1. Contexto físico y representación digital	9
2.3 Computación móvil aplicada a la educación	10
2.3.1. Paradigmas educativos tradicionales y modernos	10
2.3.2. Aprendizaje Móvil	11
2.3.2.1. Herramientas para aprendizaje móvil. Aspectos a tener en cuenta	12
2.3.2.2. El rol de los profesores	13
2.3.3. Aprendizaje Ubicuo	14
2.3.4. Aprendizaje Social y Educación Cultural	15
2.3.5. Aprendizaje Colaborativo	15
2.3.6. Aprendizaje fuera del aula	16
3. Juegos Educativos Móviles	18
3.1. Introducción	18
3.2. Educar Jugando	18
3.2.1. Historia del juego en educación	19
3.2.2. Juegos Educativos	19
3.3. Juegos colaborativos	21
3.4. Juegos Móviles	22
3.5. Ejemplos de Juegos Móviles Colaborativos	23
3.5.1. Frequency 1550	24
3.5.2. HasleInteractive Mobile Urban Drama	26
3.5.3. Savannah	29
4. Modelo de Juego Colaborativo Móvil	31
4.1. Modelo Propuesto	31
4.2. Caracterización General del Modelo	33
4.3. Aspectos Colaborativos del Modelo	38
5. Implementación del Prototipo	46
5.1. Arquitectura	46
5.2. Dificultades técnicas en el desarrollo	56
5.3. Administración Web	57
6. Simulación de un Juego Educativo Móvil Colaborativo	70
7. Conclusiones y Trabajos Futuros	101

7.1. Conclusiones	101
7.2. Trabajos Futuros	102
8. Bibliografía	105

1. Introducción

En este capítulo se hará una introducción acorde a los temas que se abordarán en esta tesina, comenzando por la motivación, los objetivos del presente trabajo y finalizando con una breve descripción de cada uno de los capítulos que hacen a la organización de la tesina.

1.1 Motivación

Entendemos que nos encontramos ante una especie de revolución digital, donde es continua la aparición de nuevos y poderosos dispositivos móviles, lo que posibilita un acercamiento de la tecnología al alcance de todos.

Los juegos utilizando computadoras como herramientas tienen un importante papel en el proceso de enseñanza-aprendizaje [4]. Teniendo en cuenta además, una colaboración entre las partes involucradas para obtener conocimiento, es decir, la actividad de pequeños equipos de trabajo, dentro de los cuales los participantes intercambian información y trabajan en una tarea hasta que todos sus miembros la han entendido y terminado, aprendiendo a través de la colaboración [1].

El aprendizaje colaborativo permite a los participantes intercambiar información, así como generar ideas, simplificar problemas y resolver tareas [2]. Echols [3] propone el concepto de aprendizaje social, un concepto basado en la idea de que el conocimiento puede crearse a través de la interacción y la colaboración de individuos, de esta forma, facilitar un aprendizaje colaborativo como el propuesto pone en manifiesto el comportamiento heurístico que fomentará la aplicación.

En cuanto a los juegos educativos, como señala De Freitas et al., “Los entornos de aprendizaje lúdicos pueden proporcionar un marco de aprendizaje interdisciplinario motivador, crear oportunidades para mejorar las aptitudes colaborativas de los estudiantes, así como ayudarles a aprender nuevos conceptos y a sintetizar información nueva” [4].

Nuestra principal motivación por tanto, esta puesta en presentar un modelo general que permita representar los principales conceptos involucrados en un Juego Colaborativo Móvil. Para probar dicho modelo se desarrollará un prototipo, utilizando y fusionando herramientas tecnológicas vigentes sin perder de vista el propósito educativo.

Se nos presenta un gran desafío al toparnos con tecnologías modernas y en auge, algunas con las que estamos familiarizados y otras con las que no. A partir del crecimiento del desarrollo y del diseño web, lo conocido como web 2.0 y el auge de los dispositivos móviles, se nos abre un gran abanico de posibilidades para el desarrollo propuesto. Tenemos, entonces, que tomar decisiones en lo que respecta a arquitecturas, frameworks de desarrollo, geolocalización, motores de base de datos, plataformas y lenguajes para desarrollo en dispositivos móviles, entre otros aspectos.

Otro enfoque motivacional es el desafío que provoca la construcción de un software para dispositivos móviles de propósito educativo colaborativo que realmente cumpla sus objetivos. Aquí es donde debemos poner énfasis y realizar una investigación bibliográfica sobre los propósitos del aprendizaje colaborativo y su puesta en práctica en el campo de los sistemas informáticos, particularmente en este caso, su funcionamiento con dispositivos móviles como herramienta para llevarlos a cabo.

1.2 Objetivos

El objetivo esencial de esta tesina está centrado en el diseño y desarrollo de un prototipo de una aplicación educativa móvil. En este marco, el objetivo específico es realizar un juego colaborativo móvil, es decir, que sea resuelto en grupos que colaboran para llegar a un resultado final.

Además del propósito colaborativo, aparece en el presente trabajo un componente esencial, el posicionamiento. Una característica importante es la movilidad, ya que la posición física determina o restringe el comportamiento y el accionar de los grupos. La información espacial juega un papel importante en nuestra sociedad. “Muchas de las actividades y decisiones humanas tienen un componente geográfico. En general, el valor de la información aumenta cuando está conectado a una ubicación” [5].

Se modelarán los conceptos relevantes involucrados en un juego colaborativo móvil, para luego a partir del mismo llevar a cabo un prototipo específico.

1.3 Organización de la tesina

Capítulo 2. En este capítulo se introducirán los conceptos relacionados con la computación móvil en general, como la computación ubicua y pervasiva, una introducción al contexto en aplicaciones móviles; así como conceptos relacionados en particular a la computación móvil aplicada en el área de educación.

Capítulo 3. Se estudiará el concepto de juegos educativos colaborativos en profundidad. Se hará un repaso de la historia de los juegos en la educación exponiendo los hitos más importantes. Luego se abordarán los conceptos de juegos educativos y juegos colaborativos en detalle para dar un cierre al marco teórico de la tesina. Finalmente se darán ejemplos de juegos educativos colaborativos similares al prototipo propuesto.

Capítulo 4. Se presentará en detalle el modelo del juego propuesto junto con una explicación detallada de las interacciones de las clases más relevantes. Se incluirán diagramas de instancia de las clases más importantes así como diagramas

de secuencia de los métodos más complejos, que ayudarán a comprender su comportamiento.

Capítulo 5. Se describirá en detalle la arquitectura elegida, el modelo y la dinámica del juego. Se detallarán las dificultades técnicas encontradas durante el desarrollo y se explicará el funcionamiento del módulo de configuración web del juego.

Capítulo 6. Se presenta la simulación de un ejemplo del juego colaborativo móvil.

Capítulo 7. Se enuncian las conclusiones que se pudieron obtener a partir del trabajo realizado, como así también los trabajos futuros que se podrían realizar.

2. Background

Hemos presenciado en estos últimos años un avance notorio en lo que a tecnología se refiere, la capacidad de cómputo de los procesadores, su tamaño cada día más pequeño, con menor consumo y menor costo asociado. La diversidad de dispositivos asociados a estos avances es notoria, desde computadores personales, tales como notebooks, netbooks, pasando por tablets, teléfonos celulares, hasta llegar incluso a los dispositivos de hardware asociados a vehículos, edificios, electrodomésticos, etc.

En este capítulo se profundizará sobre los conceptos relacionados con Computación Móvil en general y en particular en el área de educación.

2.1 Computación Móvil

En la bibliografía no hay aún una definición unificada de Computación Móvil, a continuación se presentan algunas definiciones acorde a diferentes autores.

De acuerdo a Lyytinen y Yoo 2002 (en [6]) *“La computación móvil trata fundamentalmente de incrementar la capacidad de mover físicamente los servicios computacionales con nosotros. Como resultado, la computadora se convierte en un dispositivo omnipresente que expande nuestras capacidades para inscribir, recordar, comunicar y razonar independientemente de la ubicación del dispositivo.”*. En la Figura 1 se puede observar cómo cambia el nivel de integración y movilidad a medida que nos acercamos a los conceptos de computación ubicua, móvil y pervasiva que veremos en detalle más adelante en este capítulo. La Computación Móvil tiene como característica un alto grado de movilidad y un bajo nivel de integración con el medio. Si se incrementa el nivel de integración, y se decrementa el de movilidad, se obtiene lo que se denomina una computación pervasiva, en donde los dispositivos utilizan la información de su entorno para construir modelos dinámicos. La computación ubicua requiere en cambio un alto nivel de movilidad y de integración. En su forma definitiva se refiere a cualquier dispositivo informático, que mientras se mueve con nosotros, puede crear modelos de forma incremental dinámicos de sus diferentes entornos y configurar sus servicios en consecuencia [23].

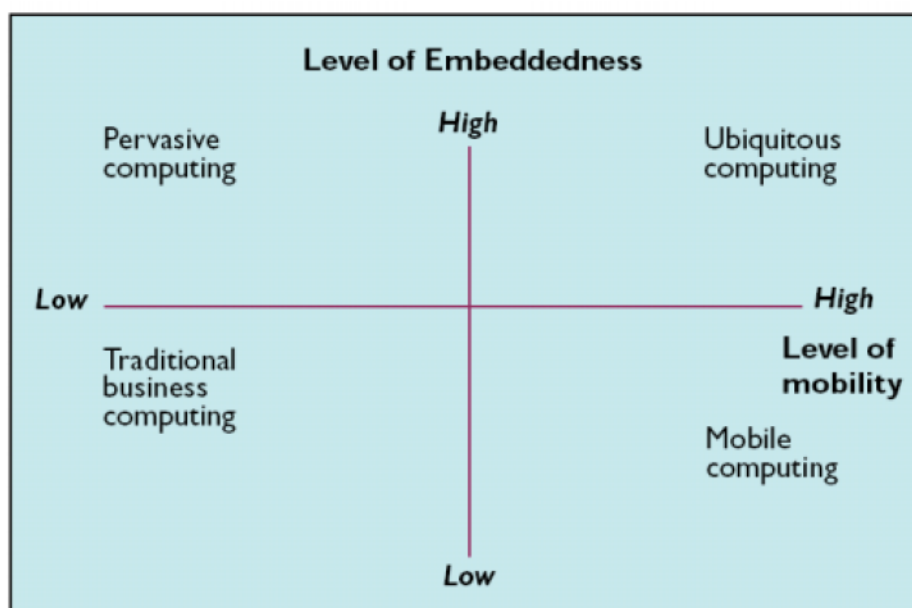


Figura 1: Comparación de los niveles de integración y movilidad entre tipos de computación [6]

Roy *et al* 2003 (en [7]) añade al concepto el rol de la comunicación inalámbrica: *“La computación móvil se puede definir como aquella que se produce mientras un usuario es capaz de moverse o se mueve alrededor, usando un dispositivo (a menudo de mano) dirigido por el usuario que es independiente de la ubicación y se puede comunicar a través de un canal inalámbrico.”*

Finalmente Talukder *et al* 2010 (en [9]) la define como: *“Computación Móvil puede definirse como un entorno informático sobre la movilidad física. El usuario de un entorno informático móvil será capaz de acceder a los datos, información u otros objetos lógicos desde cualquier dispositivo en cualquier red mientras está en movimiento.”*

Como parte del abanico de posibilidades que brinda la computación móvil, se pueden crear diferentes tipos de aplicaciones móviles para dominios muy variados, por ejemplo, turismo, educación, marketing.

Computación Pervasiva

El concepto **Computación Pervasiva** implica que la computadora tiene la capacidad de obtener información del ambiente donde opera y utilizarla para construir modelos computacionales de manera dinámica. Este proceso es recíproco, pues el ambiente también debería ser capaz de detectar a aquellos dispositivos que están entrando en él. De esta manera ambos, dispositivo y ambiente, son conscientes uno del otro y pueden interactuar de manera inteligente para brindar una mejor calidad en los servicios. Esto es el núcleo de la idea de la computación pervasiva: Un área

poblada con sensores, pads y un conjunto de modelos del ambiente (físico, social, cognitivo, etc.) [10].

Como vimos en la Figura 1, la computación pervasiva implica un alto grado de integración con el ambiente, y un bajo grado de movilidad, ya que toda la información necesaria se encuentra situada en el lugar.

Actualmente, desde el punto de vista del software, uno de los mayores desafíos en el campo de la computación pervasiva es “enseñarle” al dispositivo o aplicación sobre el contexto en el cual se desempeña [10].

Computación Ubicua

Mark Weiser introduce el concepto de **Computación Ubicua** en su artículo *Some Computer Science Problems in Ubiquitous Computing* [11]. Esta se define como:

...el método por el cual se mejora el uso de las computadoras haciendo disponibles muchas de ellas a través del ambiente físico...

... un ambiente en el cual cada persona está continuamente interactuando con cientos de computadoras, situadas en las cercanías, interconectadas de manera inalámbrica. El objetivo es lograr el tipo más efectivo de tecnología que sea esencialmente invisible al usuario.

Weiser (en [11]) plantea cómo lograr que la tecnología que nos rodea desaparezca, se haga invisible, en especial la que tiene que ver con la computación.

Esto se puede lograr de la siguiente manera:

- **Computadoras más pequeñas.** Teniendo computadoras físicamente pequeñas, con un poder de cálculo razonable, es posible pensar en empotrarlas en los aparatos que usamos todos los días.
- **Computadoras más baratas.** La computación ubicua supone que muchas computadoras son utilizadas por el mismo usuario. Esta gran cantidad de computadoras, presupone un costo no un costo no demasiado elevado por unidad.
- **Conexiones inalámbricas.** Las conexiones wireless facilitan la instalación y permiten a los dispositivos conectados en red moverse libremente.

De este modo, se logra aprovechar la movilidad a gran escala que nos brinda la computación móvil, combinada con la funcionalidad de la computación pervasiva. Es decir, como vimos en la Figura 1, un alto grado de movilidad junto con un alto grado de integración con el medio ambiente.

2.2 Contexto en Aplicaciones Móviles

Desde el punto de vista del software, una de las definiciones que más se popularizó es la que utilizó Dey *et al* [14] que define el contexto como "*... cualquier información que se puede utilizar para caracterizar la situación de una entidad. Una entidad es una persona, lugar u objeto que se considera relevante para la interacción entre el usuario y la aplicación, incluyendo el usuario y las aplicaciones en sí.*". Según los autores, existen tres tipos de información que deben ser reunidos para describir cualquier contexto específico - lugares, personas y cosas.

- Presentación de información y servicios al usuario.
- Ejecución automática de un servicio para un usuario.
- Etiquetado del contexto para su posterior recuperación.

Según Abowd y Dey [14], las categorías más importantes de información contextual son cuatro:

- **Identidad:** caracteriza a la entidad con un identificador explícito, que tiene que ser único en el dominio de nombre de la aplicación.
- **Ubicación:** incluye datos de posicionamiento y orientación, así como información sobre las relaciones regionales con otras entidades (por ejemplo, entidades vecinas). Se trata de datos geográficos, así como las relaciones espaciales.
- **Estado:** contiene propiedades, que pueden ser percibidas por el usuario. Por ejemplo, para una ubicación puede ser la temperatura actual, la iluminación ambiente o el nivel de ruido. Para las personas esto se refiere a factores físicos como los signos vitales, el cansancio o la ocupación actual.
- **Tiempo:** la fecha y la hora.

El término **sensible al contexto** fue pensado y definido por Schilit *et al* (1994) en [13] para describir la ubicación, las identidades de las personas cercanas, objetos y cambios en los objetos. Actualmente ésta terminología se utiliza en diversas disciplinas de investigación; lingüistas proporcionan dos significados para él. El contexto puede ser definido como el texto en el que una palabra o un pasaje aparece y que ayuda a determinar su significado. Una segunda definición afirma que el contexto es el entorno, las circunstancias, el medio ambiente, los antecedentes, los ajustes que especifican el significado de algún evento [13].

2.2.1 Contexto físico y representación digital

La mayoría de las entidades pueden tener tanto una representación física y una representación digital, es decir, el fenómeno físico es modelado en el sistema informático. Idealmente, la relación entre la entidad física y el modelo digital debe corresponder a la comprensión del usuario de su contexto actual (representado

como el modelo conceptual o modelo de usuario). La representación de personas físicas en el modelo digital debe por lo tanto reflejar el contexto conceptual. A modo de ejemplo, supongamos un usuario utilizando un Smartphone con un sistema hipermedia sensible al contexto. Imagine que el usuario se encuentra en frente de un edificio, y el sistema debe ser consciente de la posición del usuario a través del receptor GPS integrado del Smartphone. Sobre la base de estos datos detectados desde el contexto físico, el sistema de hipermedia puede buscar su base de datos (es decir, el contexto digital) para la información pertinente a este contexto físico en particular. El contexto físico, es decir el ambiente real, tradicionalmente no ha sido una preocupación en ciencias de la computación, manejado por otros campos como la arquitectura, la ingeniería, y diseño (industrial). Sin embargo, muchos trabajos en computación ubicua, computación pervasiva, y los campos de computación sensibles al contexto se han centrado en la captura de información sobre el mundo físico a través de los sistemas sensoriales y en ubicar esta información a disposición de las aplicaciones informáticas [12].

2.3 Computación Móvil aplicada a la educación

Desde hace ya varios años hemos visto como el avance de la tecnología posibilitó el acceso masivo a los dispositivos móviles, incluso en niños en edad escolar. Este dejó de ser un privilegio para ser más bien una consecuencia del mundo digital en el que vivimos. Si a este factor le sumamos la evolución de la conectividad inalámbrica, tenemos entonces una herramienta muy valiosa que se puede utilizar en entornos educativos, impulsando y mejorando los procesos de aprendizaje.

Nos enfrentamos a una sociedad en constante cambio, que se mueve a gran velocidad, y que nos exige como individuos vivir en un proceso de aprendizaje continuo no sólo para nuestro desempeño profesional sino también para el pleno desarrollo de nuestra vida cotidiana. Nos vemos obligados a adaptarnos a situaciones cambiantes en todos los ámbitos y a adoptar nuevos conocimientos y competencias para hacer frente a dichos cambios. Los avances tecnológicos intentan dar respuesta a las necesidades que plantea esta sociedad. De este modo, surgen tecnologías móviles que facilitan el acceso a la información y la comunicación [21].

2.3.1 Paradigmas educativos tradicionales y modernos

Según Prensky (2001) y otros (por ejemplo, Beck y Wade 2006; Klopfer 2008) en cierta medida, la educación dejó de cumplir con las necesidades de determinados alumnos. Una de las razones de este fracaso es que existe una nueva generación de alumnos que en gran medida está siendo educado con viejos métodos y paradigmas. La generación actual de alumnos está creciendo con la

información y la comunicación (TIC) incrustada en su vida diaria. Dichos alumnos manejan información digital sobre una base diaria, se conectan entre sí a través de las tecnologías móviles, funcionan de forma interactiva, a menudo realizan varias tareas más o menos al mismo tiempo y juegan juegos digitales en mayor medida que las generaciones anteriores (Beck y Wade 2006 ; OCW 2007) [16].

Debemos pensar la educación como una actividad individualizada, que puede tener lugar en cualquier momento y lugar, no necesariamente solo en el horario y ámbito escolar. Los contenidos deben ser ubicuos y, en la medida de lo posible, personificados para que de este modo cada estudiante pueda aprender y obtener las respuestas a preguntas básicas relacionadas a sus propios intereses [22].

La tecnología móvil ofrece a los estudiantes el acceso al aprendizaje siempre disponible, en cualquier momento y en cualquier lugar a través de los teléfonos inteligentes y tablets, con herramientas adicionales, incluyendo mayor funcionalidad, gráficos sofisticados, pantallas más grandes, dispositivos de audio y video, etc. que permiten a los estudiantes participar del proceso de aprendizaje, con sus materiales tradicionales y los tecnológicos, de maneras innovadoras, convincentes y motivadoras [22].

Cabe destacar también que, una de las ventajas del mundo digital es la capacidad de integrar la evaluación dentro de las herramientas de aprendizaje. Los dispositivos móviles permiten métricas detalladas sobre cómo los estudiantes se acercan a los objetivos, medir los procesos mediante los cuales se adquieren los conocimientos, y la rapidez con que recogen los conceptos y nuevas habilidades. Cuando el contenido se entrega en formato digital, es fácil implementar pruebas que evalúen la comprensión y el conocimiento rápidamente [22].

2.3.2 Aprendizaje Móvil

Si nos remitimos a su definición, el aprendizaje móvil, o m-learning, se refiere a la metodología de aprendizaje mediante el uso de dispositivos móviles con conectividad inalámbrica. Según Crompton [20] el aprendizaje móvil es el aprendizaje en contextos múltiples, a través de interacciones sociales y de contenido, mediante el uso de dispositivos electrónicos personales. Se puede comprender al aprendizaje móvil como “cualquier actividad que permite a las personas ser más productivas cuando consumen, interactúan o crean información a través de un dispositivo portátil digital compacto que el individuo utiliza con regularidad, que posee una conectividad fiable y cabe en un bolsillo o en la cartera” [Park en 23]. Desde un punto de vista socio-pedagógico, y de forma contraria al concepto tradicional de enseñanza, el m-Learning se orienta hacia el aprendizaje colaborativo, flexible, espontáneo e informal y basado en la resolución de problemas [21].

El aprendizaje móvil tiene algunas propiedades en función del usuario, su entorno y la tecnología utilizada. Es utilizado comúnmente en tareas de tiempo limitado y con entregas de contenido orientado al contexto.

Por otra parte, el aprendizaje móvil incluye la interacción multimedia utilizando varias tecnologías manejadas en el dispositivo móvil, como reconocimiento de voz, sonido, tacto, grabaciones, videos, geolocalización, etc. También puede soportar un entorno de colaboración donde los usuarios se contactan con otros para ayudarse mutuamente (por ejemplo, amigos, compañeros de estudio o incluso expertos). Tengamos en cuenta también que, para llevar a cabo el aprendizaje móvil de una manera satisfactoria, las cualidades de un dispositivo móvil que se utilicen en aplicaciones orientadas al aprendizaje, deben ser consideradas de manera muy precisa [13], por ejemplo un receptor gps que no tenga la precisión adecuada para una aplicación que lo precisa, puede llevar a que los alumnos pierdan motivación o incluso a que la aplicación no cumpla su propósito.

2.3.2.1 Herramientas para Aprendizaje Móvil, aspectos a tener en cuenta

Las herramientas utilizadas para actividades que tengan que ver con m-learning deben ser portátiles (es decir, ligeras, robustas y alimentadas por batería), que puedan utilizarse al estar de pie o caminando (eliminando las computadoras portátiles normales como plataformas potenciales), que posean navegación de apoyo (GPS), facilidad para introducir texto así como sea posible, también en muchos casos debe proveer la entrada/salida de multimedia (captura de una fotografía o grabación de sonido/vídeo), tener conectividad inalámbrica, y ser capaces de asociar la ubicación con los datos de alguna manera.

Hay que tener en cuenta también que una alta portabilidad implica un tamaño pequeño, lo que restringe el tamaño de la pantalla y dificulta la capacidad de uso.

Enumerando las características tecnológicas principales asociadas al m-learning serían [21]:

- Portabilidad, debido al pequeño tamaño de los dispositivos.
- Inmediatez y conectividad mediante redes inalámbricas.
- Ubicuidad, ya que se libera el aprendizaje de barreras espaciales o temporales.
- Adaptabilidad de servicios, aplicaciones e interfaces a las necesidades del usuario.

También, llegado el caso, existe la posibilidad de incluir accesorios como teclados o lápices para facilitar su uso, aunque con los últimos avances tecnológicos cada vez es menos necesario.

En cuanto a la conectividad, en los últimos años, la mejora en las infraestructuras de redes inalámbricas es a la vez causa y efecto del impacto de las tecnologías móviles en todos los ámbitos de la vida cotidiana. La proliferación de redes inalámbricas da respuesta a las necesidades sociales e individuales de conectividad, movilidad y flexibilidad. Mucho han evolucionado las redes inalámbricas en pocos años, con velocidades de transmisión cada vez mayores,

mayor seguridad y mayor cobertura [21].

También no hay que dejar de tener en cuenta que estos dispositivos tienen algunas deficiencias, que pueden superarse fácilmente si la arquitectura de la plataforma de aprendizaje móvil que se desarrolla es consciente de las mismas. Un dispositivo móvil tiene una cantidad específica de energía que se descarga muy rápidamente cuando se utiliza junto con la comunicación de red. El ancho de banda limitado que es proporcionado por las empresas de telecomunicaciones es también una brecha que se tiene que considerar, ya que grandes cantidades de datos no se pueden transferir de manera fácil y tampoco rápidamente en el tiempo. Seguramente los datos tengan que ser transferidos en tiempo real y como los intervalos de uso de la plataforma de aprendizaje móvil son frecuentes pero cortos, es probable que el usuario no pueda mantenerse a la espera del envío de datos por un largo período de tiempo [13].

2.3.2.2 El rol de los profesores

El aprendizaje móvil abre un abanico de posibilidades para la enseñanza moderna. Permite a los profesores buscar métodos de aprendizaje con herramientas que son afines a los alumnos, en las que se sienten cómodos y están habituados a utilizar. Un desafío constante para los profesores es tener alumnos motivados. Un alumno motivado es fácil de describir: entusiasta, centrado y comprometido. Un alumno motivado muestra un claro interés en lo que se está realizando, se esfuerza mucho y persiste en el tiempo (Garriset en [16]). Según Schwabe y Goth (en [16]) la tecnología permite la inmersión en una realidad mixta y por lo tanto proporciona experiencias de aprendizaje altamente motivadores.

La duración fija de las lecciones en clase puede llegar a ser una limitación para la planificación e implementación del aprendizaje innovador, tal como lo son los juegos que veremos en el próximo capítulo. La falta de familiaridad con alguna de estas prácticas puede repercutir en el éxito o fracaso de las mismas. Es necesario cierta flexibilidad en términos de la duración del aprendizaje, los contextos en donde se realiza el mismo, así como también el tiempo necesario para la preparación de la lección y un buen apoyo técnico [25].

Acorde con West (en [22]), los docentes son cruciales para el éxito de cualquier reforma de la escuela o el aprendizaje innovador. Como agentes de aprendizaje en el aula, guían la educación, de manera que son muy importantes. Si los maestros cuentan con la formación y la financiación necesaria para emplear herramientas útiles, van a ser más eficaces en la transmisión de conocimientos y habilidades a sus estudiantes.

La tecnología digital ayuda a los maestros a reflexionar sobre nuevos modelos de dictado de clases. Los estudiantes pueden tomar más responsabilidad de su propio aprendizaje, mientras que los profesores pueden concentrarse en la resolución de problemas y en la construcción de habilidades críticas más avanzadas para las personas en sus aulas. El resultado será una colaboración educativa más

satisfactoria para los estudiantes y maestros [22].

2.3.3 Aprendizaje Ubicuo

Aprendizaje ubicuo (u-learning) es un enfoque innovador que integra las tecnologías inalámbricas, los teléfonos u otros dispositivos, y la sensibilidad al contexto para detectar la situación de los alumnos en el mundo real y proporcionar apoyo adaptativo o guía en consecuencia, es decir, proporcionar actividades de aprendizaje accesibles en cualquier momento y lugar [24]. A diferencia del aprendizaje móvil, en u-learning no son imprescindibles los dispositivos móviles.

A pesar de que se pueden utilizar los dispositivos móviles, se pueden también utilizar las computadoras personales, televisores, etc. En general, el término u-learning se distancia un poco del dispositivo específico, se centra en el acceso en diferentes contextos y situaciones, consiste en aprender en un ambiente donde "todos los estudiantes tienen acceso a una variedad de dispositivos y servicios digitales, incluidos los ordenadores conectados a Internet y dispositivos móviles, cuando y donde lo necesiten" según Van't Hooft (2007 en [13]).

Algunas de las principales características de u-learning en un enfoque dirigido hacia el alumnado son:

- **Permanencia:** Con la utilización de la tecnología pertinente, los alumnos nunca pierden sus trabajos a menos que se eliminen a propósito o debido a alguna falla de hardware. Además, todos los procesos de aprendizaje se registran continuamente cada día.
- **Accesibilidad:** Los alumnos tienen acceso a sus documentos, datos o videos desde cualquier lugar. Esa información se ofrece en base a sus peticiones. Por lo tanto, el aprendizaje en cuestión es auto dirigido.
- **Inmediatez:** En cualquier lugar que estén que los alumnos, pueden recibir información de inmediato. De este modo, los alumnos pueden resolver problemas rápidamente. En caso contrario, el alumno puede grabar las preguntas y buscar la respuesta más tarde.
- **Interactividad:** Los estudiantes pueden interactuar con los expertos, maestros o compañeros, en forma sincrónica o asincrónica. Por lo tanto, la ayuda es más accesible y el conocimiento se vuelve más disponible.
- **Situando las actividades de instrucción:** El aprendizaje puede ser introducido en nuestra vida cotidiana. Los problemas que se plantean, así como los conocimientos necesarios son presentados en sus formas naturales y auténticas.
- **Adaptabilidad:** Los alumnos pueden obtener la información correcta en el lugar correcto de la forma correcta.

De este modo, u-learning es un tipo especial de enseñanza que se individualiza, amplía el acceso, permite el trabajo colaborativo, y puede dar una respuesta instantánea a los estudiantes. En comparación con el trabajo en las computadoras de escritorio inmóviles únicamente, u-learning permite que el ambiente de aprendizaje sea visitado en cualquier lugar y en cualquier momento.

[13].

Una gran diferencia entre el aprendizaje situado en el ambiente de una computadora personal o en el aula y el aprendizaje utilizando dispositivos móviles, es el contexto en el que el usuario está. Los dispositivos móviles poseen algunas herramientas para capturar cierta información de fondo que puede ser útil para personalizar la experiencia de aprendizaje. El concepto contexto se discutió con cierto detalle en la primer parte del capítulo [13].

2.3.4 Aprendizaje Social y Educación cultural

La educación cultural es vista principalmente como un proceso creativo que no trata a los alumnos sólo como consumidores pasivos con habilidades normales, sino que los ve como productores activos de conocimientos y medios audiovisuales que se centran en la innovación y la creatividad.

Mediante el uso de medios digitales, como los juegos de computadora, teléfonos celulares, Internet y la tecnología de localización (GPS), la educación cultural se acerca más a los estilos de vida de los alumnos y, por lo tanto, se convierte en un proceso más atractivo para ellos (estilo de vida digital). Los medios digitales también se utilizan para estimular la colaboración entre los alumnos y sus compañeros, fomentando así, en lugar de reducir, el proceso de aprendizaje social.

Los individuos se sienten conectados y parte de su propio aprendizaje, el aprendizaje a distancia ya no es tal, perdiendo la dimensión espacial para ser sólo aprendizaje y participación. Ya no son meros alumnos espectadores en una sociedad del espectáculo, ahora los discentes han tomado posesión del entorno virtual y construyen colaborativamente los conocimientos [21].

Mientras están jugando y discutiendo tácticas de juego, por ejemplo, los alumnos están usando y desarrollando sus habilidades de comunicación y colaboración. Según Paul Kirschner, es una forma de aprendizaje dependiente. Esta es "la adquisición de competencias que incluyen no sólo el conocimiento individual y las habilidades cognitivas, sino también las actitudes y las habilidades interpersonales (es decir, social y afectivas). El proceso de aprendizaje es un proceso de grupo que lleva a los logros del grupo y se valora como tal" [17].

Según Riviére [18], el aprendizaje social parece especialmente adecuado para explicar la influencia que pueden tener en el desarrollo humano las nuevas tecnologías de la comunicación, que presentan modelos cada vez más diversos y lejanos del contexto espacial y temporal inmediato de las personas. Estos medios inducen experiencias sociales a través de modelos que van mucho más allá de los proporcionados directamente por la familia y la escuela. Sin embargo, del mismo modo que los individuos no son simples marionetas de los impulsos inconscientes ni de las contingencias ambientales en esta teoría, tampoco lo son de las influencias sociales que los modelos ejercen sobre las personas.

2.3.5 Aprendizaje Colaborativo

El término aprendizaje colaborativo hace referencia a metodologías de aprendizaje que surgen a partir de la colaboración con grupos que comparten espacios de discusión en pos de informarse o de realizar trabajos en equipo. El trabajo grupal apunta a compartir la autoridad, a aceptar la responsabilidad y el punto de vista del otro, a construir consenso con los demás [26]

En pos de un aprendizaje colaborativo, en el ámbito educativo, el docente debe diseñar la propuesta, dividir las tareas, probablemente asumir el rol de mediador cognitivo, proponiendo pautas esenciales que apunten a la construcción de conocimiento y no a una repetición de información obtenida. Finalmente reunir los logros obtenidos, compartir los resultados con la clase, y resolver cuestiones pendientes individuales o grupales que hayan quedado sin resolver [26].

Cuando los alumnos trabajan colaborativamente tienen que negociar metas, representar el problema, como también interpretar el significado de conceptos y procedimientos involucrados. Tienen que hacer explícitos sus conocimientos y pensamientos. Para establecer la comprensión común, deben intercambiar argumentos y negociar significados. La interacción social entre compañeros proporciona una situación ideal para que el proceso de aprendizaje se desarrolle oportunamente [26].

2.3.6 Aprendizaje fuera del aula

La mayoría de las escuelas han dedicado tiempo, dinero y esfuerzo en construir, mejorar y mantener salas de computación, pero se vieron imposibilitadas de mover la información digital fuera de las aulas de informática. Era muy difícil y costoso proveer al alumnado de dispositivos electrónicos móviles hace algunos años. Era necesario una gran inversión e infraestructura para posibilitar al alumnado un aprendizaje fuera del aula, los dispositivos eran muy costosos, grandes, pesados, sus baterías no duraban lo suficiente y a menudo se necesitaban dos o tres dispositivos por alumno para proveer portabilidad, conectividad, navegación y entrada de texto por ejemplo. Con el avance de la tecnología esta brecha se redujo considerablemente, actualmente un pequeño dispositivo puede proveer todo este conjunto de herramientas y muchas veces es el propio alumno el que lo facilita.

Estamos ante un nuevo paradigma de aprendizaje en relación al modo de acceso, ubicuo y no necesariamente dentro del contexto del aula. Asistimos a una evolución del término “distancia” en el ámbito de la educación. Distancia “implica un cambio geográfico entre donde residen los contenidos y el lugar en el que se toman, manteniendo siempre una conexión física entre ellos”. En cambio en el aprendizaje móvil el término distancia implica que “la recuperación o el acceso al contenido puede hacerse en movimiento, sin importar el lugar y obteniendo un mayor provecho del tiempo disponible”.(Beretta, citado por Morales en [21])

Las tecnologías móviles basadas en la localización ofrecen la oportunidad de integrar el aprendizaje en entornos auténticos y con ello potenciar la participación y el aprendizaje fuera de los entornos educativos formales tradicionales. Con los

dispositivos móviles, es posible mezclar datos virtuales con datos del mundo real (es decir, lugares y contextos) y de ese modo conectar un mundo virtual a la vida real (Klopfer y Squire) [16]. Las excursiones son a menudo una parte integral de este tipo de proyectos y los estudiantes se benefician enormemente de este aprendizaje ya que les permite relacionarse con los entornos del mundo real [12]. El desarrollo de la tecnología omnipresente y móvil hace posible la combinación de la información digital con el entorno físico y permite a los estudiantes llevar un contexto digital con ellos.

Al mover algo de la educación fuera de las aulas, las lecciones aprendidas pueden ser mejores, ya que tiene lugar la experiencia en el lugar además de la lectura en clase. El aprendizaje se convierte en esencial para llevar esta experiencia de nuevo al salón de clases para futuros trabajos y discusiones.

Además, mucha información está situada, es decir, puede estar vinculada a lugares y circunstancias específicas. Al moverse en un paisaje real, el estudiante de manera simultánea puede navegar por una información virtual en el que los datos pertinentes a la ubicación actual se presentan automáticamente. Esto permite la presentación de información sobre la ubicación actual, y también abre las posibilidades para las preguntas “in situ” (ej “¿Qué edad tiene la iglesia que está frente nuestro?”). Nos induce a buscar y navegar en el contexto y en el aprovechamiento de los conocimientos sobre la hora, lugar y propósito en la investigación de la información en el campo. La búsqueda y la navegación en el sitio se pueden mejorar considerablemente mediante el uso de la tecnología móvil y sensible al contexto [12].

3. Juegos Educativos Móviles

3.1 Introducción

En este capítulo pondremos el foco en los juegos educativos móviles como método de aprendizaje colaborativo móvil. Es decir, a grandes rasgos, el juego como herramienta teniendo como objetivo el logro del aprendizaje, a través del uso de dispositivos móviles. Aprovechando, con ello, todas las ventajas descritas en el capítulo anterior, tales como el aprendizaje situado, activo, reflexivo, individual y colectivo. Pondremos énfasis en el aspecto colaborativo de los juegos educativos y en cómo éste ayuda a la construcción de un conocimiento persistente.

El uso de juegos móviles en la educación combina el aprendizaje situado y activo con la diversión de una manera potencialmente excelente. Es muy posible que la enseñanza basada en juegos pueda abordar de manera más adecuada la forma en que los jóvenes aprenden hoy en día y hacerlos participar con más éxito en el aprendizaje significativo que con los métodos tradicionales de aprendizaje (Prensky, Gee, Shaffer, van Eck) [16].

Los niños pasan sus días jugando y divirtiéndose, sin darse cuenta a veces que están construyendo conocimientos y aprendiendo nuevos conceptos o formas de relacionarse. Ellos juegan por instinto, se mueven, corren, se pelean, se amigan, trepan, saltan. Ponen en práctica sus habilidades físicas y mentales. "El juego es toda actividad que está dotada de placer funcional, y que se mantiene en pie en virtud de este mismo placer y gracias a él, cualesquiera que sean su ulterior rendimiento y sus relaciones de utilidad" (Buhler Carlos). La definición de Dempsey dice que un juego es "un conjunto de actividades que afecte a uno o más jugadores. Tiene metas, limitaciones, beneficios y consecuencias. Un juego es orientado por las normas y artificial en algunos aspectos. Por último, un juego consiste en algunos aspectos de la competencia, incluso si esa es una competencia con uno mismo [16].

Según Ritterfeld & Weber, "tanto la motivación como el paradigma de fortalecimiento se basan en un concepto aditivo de entretenimiento y educación, en el que se añade información educativa a un programa de entretenimiento o viceversa (...) Sin embargo (...) el entretenimiento educativo desarrolla su máxima estrategia potencial si la información proporcionada se convierte en una parte esencial de la experiencia" [17].

Tener alumnos motivados es lo que desean muchos maestros. De hecho, un alumno motivado es fácil de describir: entusiasta, centrado y comprometido. Un alumno motivado muestra un claro interés en lo que él o ella está haciendo y le gusta lo que él o ella está haciendo, se esfuerza mucho y persiste en el tiempo [16]

3.2 Educar Jugando

3.2.1 Historia del juego en educación

Según Stephen Heppell (2006) en la década del '80, las generaciones estaban cautivadas por las pantallas de televisión, en ese entonces eran meros espectadores pasivos y los juegos relacionados se limitaban a “golpear platos con ladrillos”. Las generaciones futuras se vieron cautivadas también por las pantallas, pero en un sentido más amplio, se interesaron en el uso y control de las mismas, pasaron de una generación pasiva a una plenamente activa. Allí aparecieron los pioneros en lo que respecta a juegos, trabajaron junto a las primeras generaciones de lenguajes de programación intentando realizar mejores juegos, teniendo en cuenta la resolución de problemas, utilizando el ingenio, los desafíos tecnológicos, avanzando a pasos agigantados generando toda una industria creativa. Llegamos al siglo 20 con una industria del juego en auge, colaborando, en constante desafío, valiéndose del ingenio y la innovación. Del otro lado, existía un sistema educativo que no iba de la mano con los avances, no existían prácticamente los juegos educativos, había una actitud pasiva, algo conformista al respecto [25].

Con el nuevo siglo llegaron los cambios, los individuos ligados a la enseñanza notaron que a través de los juegos, los alumnos se enfrentaban muchas veces a desafíos intelectuales, que podían generar nuevos conocimientos y que los formaba de manera individual y en conjunto al trabajar colaborativamente en algunos casos. La educación entonces comenzó a tomar en serio a la industria del juego, y la industria a su vez, comenzó a dedicarle más esfuerzo al desarrollo de los mismos, notando que había un mercado y un interés real en este nuevo estilo de aprendizaje. Se creó como una especie de simbiosis donde las dos partes se necesitaban mutuamente [25].

Los primeros juegos y simulaciones, para fines específicamente educativos, se encontraban en los juegos de guerra (Coleman 1971). Sin embargo mientras que las simulaciones se consideran aceptables herramientas de formación, juegos asociados con la violencia y las actividades de ocio han sido más ampliamente resistidos por los profesores y padres por igual [25].

Finalmente, en estos últimos años, el juego fue despegándose de la dependencia de una computadora frente a un escritorio o de un sillón frente a una pantalla. Las nuevas formas de juego, tales como los juegos móviles o de realidad aumentada por ejemplo, implican que el juego deba o pueda desarrollarse en contextos exteriores, con todos los desafíos y nuevas posibilidades que esto conlleva [25].

3.2.2 Juegos Educativos

No es sencillo dar una definición concreta y abstracta del juego educativo, esto es en gran parte debido al hecho de que los juegos son inherentemente multidisciplinarios. Juul (2003) y otros (Salen y Zimmerman, 2004) han discutido el

tema de definir los juegos que definen su alcance, y no existe un consenso real entre las disciplinas en términos comunes y sus definiciones. Por ejemplo, los juegos se han definido como juegos basados en reglas (juegos, 1978; Salen y Zimmerman, 2002) o como una actividad voluntaria y libre (Huizinga, 1980; Caillois, 1961) fuera de la vida "normal" [25].

Freitas en su revisión de aprendizaje basado en el juego (2006) define al juego educativo como: *"Las aplicaciones que utilizan ciertas características del hardware de dispositivos ya sean móviles, computadoras de escritorio o juegos de ordenador, para crear experiencias de aprendizaje interesantes, motivadoras, lúdicas y envolventes para la entrega de objetivos de aprendizaje específicos, resultados y experiencias."* [25].

Esta definición tiene como objetivo abarcar las características inmersivas e interesantes de los juegos con la eficacia educativa para fomentar el desarrollo de aplicaciones de aprendizaje que pueden apoyar los diferentes sectores de aprendizaje, al tiempo que proporciona un lenguaje que sea comprensible y suficientemente neutral para diferentes comunidades de partes interesadas, tales como estudiantes, profesores, organizaciones de formación, instituciones educativas y responsables políticos [25].

Como hemos señalado en el capítulo anterior, la motivación es un aspecto clave de un aprendizaje efectivo. Cuando los juegos están involucrados debe ser sostenida a través de respuestas de retroalimentación, reflexión y participación activa para lograr su objetivo y generar conocimiento en el participante. Un reto clave para los diseñadores es, entonces, conseguir el equilibrio correcto entre el juego como entretenimiento y el cumplimiento de los resultados del aprendizaje [25].

El éxito de los juegos en los contextos educativos no es una ciencia exacta, ha habido una escasez de estudios científicos que analicen su uso. Además de la confusión sobre lo que constituye un juego de aprendizaje sigue siendo problemática y los variados tipos y géneros de juegos han añadido el reto de la diversidad. Según entendemos y notamos a lo largo de este trabajo, existe una necesidad de estudios de referencia más rigurosos que puedan cuantificar cuánto y de qué manera los juegos y simulaciones en la actualidad se están utilizando con mayor eficacia para apoyar el aprendizaje [25].

Según Freitas (2006), históricamente ha habido una percepción dominante de los juegos como una actividad de ocio sin ningún valor pedagógico. La fusión entre los juegos de ocio y juegos educativos es una confusión comprensible, ya que las condiciones de ambos también se utilizan como sinónimos. Sin embargo, la confusión da lugar a ideas erróneas y a la utilización ineficaz de los juegos de ocio en las aulas. Existe una discusión y estudios que deliberan sobre si la eficacia de los juegos de ocio pueden influir en el logro de los objetivos de aprendizaje conjunto. Hay dos explicaciones principales para ello, en primer lugar que la manera en que los juegos de ocio han sido usados y probados en contextos de aprendizaje no

siempre ha seguido los principios de un buen diseño de aprendizaje, es decir, que operen desde los resultados y los objetivos y no sólo se utilicen como un descanso del aprendizaje. En segundo lugar, una explicación más especulativa, que indica que el aprendizaje basado en el juego puede no funcionar de la misma manera como funcionan las actividades de aprendizaje usuales, por lo que se puede necesitar un soporte pedagógico diferente o un marco de aprendizaje con el fin de utilizar el aprendizaje basado en el juego con mayor eficacia. La percepción de los videojuegos como herramienta de aprendizaje para la educación está cambiando cada día y de a poco se van introduciendo como métodos alternativos de enseñanza [25].

Hay barreras significativas para la aceptación y puesta en práctica de los juegos en el ámbito educativo [25]. Estos incluyen:

- el acceso al hardware adecuado incluyendo PCs con gráficas de gama alta de tarjetas de video
- apoyo técnico eficaz o acceso al soporte técnico adecuado
- familiaridad con los juegos basados en software
- comunidad de práctica dentro del cual pueda gestionar orientación y apoyo
- tiempo suficiente para preparar el aprendizaje basado en el juego efectivo
- grupos de alumnos que deseen aprender el uso de enfoques basados en el juego eficaces
- costo del software de juegos educativos o licencias.

3.3 Juegos Colaborativos

La incorporación del juego en la implementación de ambientes de aprendizaje colaborativos asistidos por computadoras permite configurar modelos más activos, basados en la construcción del conocimiento, logrando de este modo mayor dinamismo y motivación de los estudiantes (Beichner, 1994; Pence, 1995-96; Wiburg, 1994) [26].

Jugar de modo colaborativo permite superar miedos, gestionar conflictos y comprenderse a uno mismo y a los demás. Los niños educados en la colaboración, la aceptación y el éxito compartido tienen muchas posibilidades de desarrollar una personalidad saludable (Jordi Miralles, 2006) [27].

Según Miralles (2006 en [27]), los juegos colaborativos pueden ayudar a los niños a:

- Tener más confianza en sus capacidades
- Tener más confianza en los demás
- Desarrollar sentimientos, expresarlos, aceptarlos, transformarlos y a aceptar los demás
- Ver a los demás como seres complementarios

- Darle espacio a los más débiles y que se sientan valorados como los demás
- Sentirse responsable de sí mismo y de los demás
- Superar miedos
- Comunicarse positivamente con los demás
- Gestionar los conflictos
- Comprenderse a sí mismo y a los demás

Un aspecto muy importante a la hora de implementar un juego educativo es tener en cuenta la comunicación entre los sujetos involucrados. La comunicación es el principal elemento necesario para que se lleve a cabo la construcción de conocimiento. Con la utilización de estas tecnologías se incrementan las posibilidades de interactuar con los miembros del grupo [21]. La disponibilidad de una herramienta de comunicación, ya sea directa (por ejemplo, texto, chat o voz) o indirecta (por ejemplo, la visualización compartida), fue un importante contribuyente a la calidad de las decisiones (Thoughtlink, 2001 citado en [25]).

3.4 Juegos Móviles

El potencial aprendizaje de las tecnologías móviles y juegos basados en la localización, radica en la posibilidad de integrar el aprendizaje en un ambiente auténtico, mejorar la participación y fomentar el aprendizaje fuera del sistema educativo oficial tradicional. Una estrategia de aprendizaje basado en el juego móvil es altamente eficaz cuando se compara con una trayectoria regular de instrucción basada en proyectos [16]. Se ponen en práctica todos los conceptos vistos en el capítulo anterior, tales como computación ubicua, pervasiva y sensible al contexto.

Los individuos que juegan a estos juegos, se encuentran generalmente muy comprometidos con el mismo: están totalmente inmersos en el juego y pueden jugar durante horas y horas con poco o ningún conocimiento del mundo que los rodea (Beck y Wade 2006; Shaffer 2006) [16].

Si nos centramos ahora en la usabilidad de los juegos móviles hay varios aspectos que debemos tener en cuenta, como resume Redwood (en [29]):

- **Botones simples:** los controles de un juego móvil deben ser lo más simples posibles. El espacio para botones en tablets y smartphones es limitado. Los juegos más populares a menudo requieren unos pocos botones y utilizan mucho los atributos de los dispositivos, como el acelerómetro y las pantallas táctiles.
- **Utilizar la pantalla de forma adecuada:** Siempre tener en cuenta que las pantallas son pequeñas, y se reducen aún más cuando existen menús desplegados, lo que puede hacer que el juego se vuelva difícil de jugar.
- **Retroalimentación:** Cuando el usuario realiza una acción, es deseable siempre que este sea notificado de alguna manera del éxito

de la misma, ya sea con un mensaje informativo, o aprovechando la vibración o los sonidos de los dispositivos.

- **Íconos claros:** es importante mantener los íconos claros en la pantalla, a menudo reflejan atajos mediante la utilización de símbolos conocidos, como el signo pesos, o una llave de herramientas. Debemos evitar la ambivalencia para no dar lugar a confusiones.
- **Administración de sonidos intuitiva:** En principio silenciar un teléfono o tablet también debería silenciar el juego, para que el usuario no tenga que perder tiempo silenciándolo de forma manual. Aunque es una característica que tienen que tener los juegos ya que a menudo el usuario puede querer escuchar la radio o escuchar música mientras juega.
- **Proteger el juego de las interrupciones o caídas:** En un teléfono móvil, recibir una llamada telefónica o un mensaje implica salir del juego para realizar otra actividad. Es deseable cuando reanudamos el juego, que este no se haya detenido, o bien que podamos volver a la situación en que nos encontrábamos (puede ser mediante el manejo de sesiones de usuario) previa a la interrupción o caída.
- **Metas y Recompensas:** Los juegos deben proporcionar objetivos claros para los jugadores y metas visibles para trabajar en el juego. Sin un propósito u objetivo definido, los usuarios perderán el interés en el juego y en seguir adelante. También es importante, si el juego lo permite, establecer niveles y al pasar un nuevo nivel premiar a los jugadores por las acciones que han realizado, de este modo se los incentiva para seguir jugando con la promesa de nuevos niveles desbloqueables.
- **Sincronicidad:** En los juegos que requieren interacción entre los jugadores, es muy importante el sincronismo entre ellos. Un sincronismo pobre, lento o con fallas hace que los jugadores pierdan motivación e interés y no se logren los resultados esperados.

3.5 Ejemplos de Juegos Móviles Colaborativos

En esta sección se presentan tres aplicaciones basadas en juegos colaborativos móviles con características similares a las propuestas en el presente trabajo -que detallaremos en profundidad en los próximos capítulos.

Las aplicaciones son:

- Frequency 1550 [16][17]
- HasleInteractive -Mobile Urban Drama- [19]
- Savannah [28]

3.5.1 Frequency 1550

Frequency 1550 es un juego de “ciudad” móvil, ciudad en el sentido que se centra y se juega en una ciudad determinada, desarrollado por la Sociedad Holandesa Waag (Waag Society), en Amsterdam en 2005. Está pensado para alumnos de 12 a 16 años en su primer año de educación secundaria de edad para que adquieran, jugando, conocimientos específicos acerca de la ciudad de Amsterdam medieval. Ayudándose con el uso de Internet, los teléfonos inteligentes y la tecnología GPS. Es entonces una interesante mezcla de juego e historia [16][17].

De acuerdo con la historia de fondo de Frecuencia 1550, el equipo de desarrollo Waag Society se encontró con algunas dificultades técnicas con la red UMTS (Sistema universal de telecomunicaciones de Móviles) de Amsterdam durante una prueba de red. Frecuencia 1550 está mágicamente interfiriendo con un período de tiempo diferente, el año 1550 de la era medieval tardía. A través de la red UMTS, el alguacil de la ciudad medieval entra en contacto con el presente, que es en el siglo 21 de Amsterdam. A través de todo tipo de malentendidos, el alguacil toma a los desarrolladores (y luego los alumnos) como peregrinos que visitan Amsterdam en 1550 con el fin de visitar la Sagrada Hostia, una reliquia especial asociada con “El Milagro de Amsterdam”. La reliquia ha desaparecido misteriosamente, y el alguacil sugiere un trato: él les dará acceso a la ciudadanía, siempre que lo ayuden a recuperar la santa reliquia. Seis equipos de cuatro alumnos elegidos al azar (Dos de ellos ubicados en la Sede HQ en De Waag, los otros dos caminando por las calles de Amsterdam) cada uno asume sus roles como peregrinos de la competencia, por lo que entran en el mundo del juego [17].

Infraestructura técnica: Los dos miembros del equipo que deambulan por Amsterdam están equipados con un GPRS Gamephone Nokia 6600 vinculado a un sistema de posicionamiento global (GPS) y un videoteléfono Sony Ericsson Z1010 UMTS. Sus dos compañeros en la Sede tienen un videoteléfono también, y una computadora portátil con conexión a Internet. Todos los teléfonos inteligentes tienen conexión a Internet y el gamephone está conectado constantemente al servidor del juego (en la sede en Waag Society) [17].

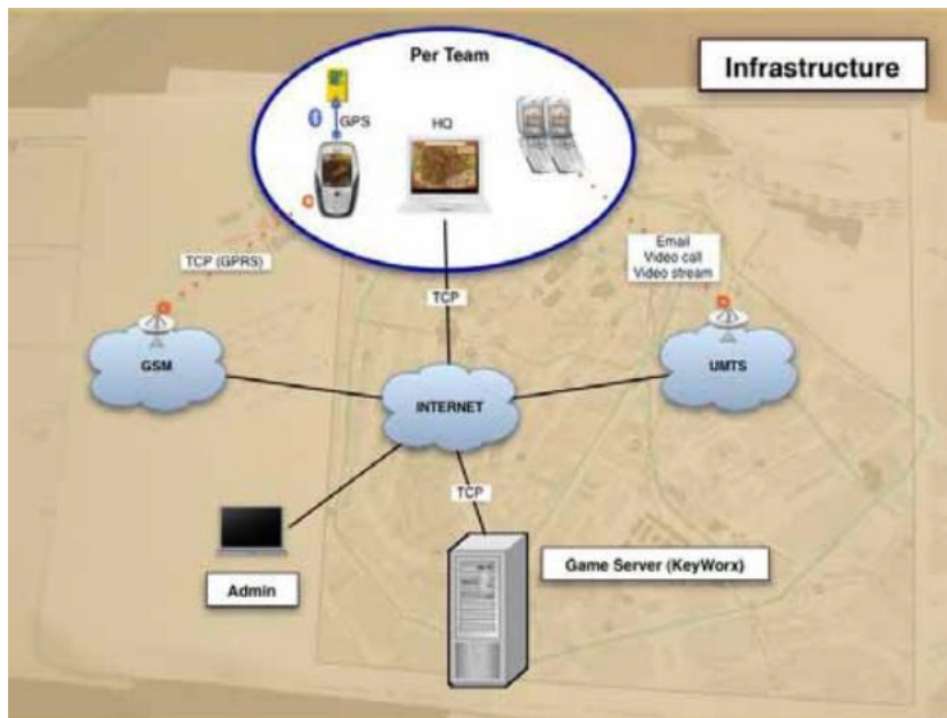


Figura 2. Infraestructura del juego Frequency 1550

Con la ayuda de un receptor GPS, el gamephone muestra la posición de su equipo en un mapa medieval de Amsterdam. A través del uso de su videoteléfono y del Gamephone, un equipo puede llamar (por video) a su cuartel general, recibir y ver mensajes de video pregrabados en los que figuran personajes medievales, y también realizar y enviar tareas (clips de video y fotos) a la Sede ubicada en De Waag. Mediante el uso de sus computadoras portátiles, los miembros del equipo en la Sede tienen una aplicación del juego con acceso a Internet para buscar información, echar un vistazo a las referencias históricas y enviar la información pertinente a los compañeros que están en la ciudad, lo que les ayuda a cumplir sus tareas asignadas [17].

Metas del juego: Cada alumno es un peregrino representando la identidad de un mendigo o un comerciante y tienen diferentes derechos y un estatus diferente. Vienen a la ciudad para ver la “Hostia del Milagro” y fundar su propio monasterio en Amsterdam. Pero en primer lugar, tiene que merecer el derecho de construir un monasterio al convertirse en un burgués (ciudadano)”. Al mismo tiempo, los equipos están compitiendo entre sí. Cada equipo de peregrinos tiene que decidir si se mantiene alejado de otros equipos o llama a una confrontación (el de mayor status vence). El equipo ganador se lleva como ganancia días de ciudadanía medievales del equipo que pierde. Los miembros del equipo también pueden soltar ratas medievales virtuales (es decir, el equivalente a una bomba virtual) en el otro equipo que hace que la pantalla del teléfono inteligente se ponga en blanco y por lo tanto deshabilita el uso de este teléfono de forma temporal [17].

Tareas: Tan pronto como un equipo de peregrinos (jugadores) ha llegado a

un cierto lugar, se envían videos automáticamente desde el servidor situado en De Waag al videoteléfono UMTS con la ayuda del receptor GPS. El equipo recibe órdenes tales como: "Visualizar el videoclip de Schreierstoren" con el fin de obtener días de ciudadanía, un equipo necesita demostrar su conocimiento de Amsterdam medieval haciendo pequeñas tareas basadas en la localización y contestando preguntas específicas sobre la historia de la ciudad, por ejemplo con respecto a Schreierstoren [17].

Se anima a los alumnos a cumplir al menos dos de las tareas por sector, con el fin de habilitar un nuevo sector con una nueva historia y lugares donde están las tareas. Ellos pueden ganar puntos por estas tareas, ayudándolos a ganar el juego. Cada tarea se inicia con un video a modo de introducción, y luego a los alumnos se les pide realizar una investigación y poner en práctica algún tipo de situación histórica que tienen que registrar y enviar a la sede con su teléfono inteligente [17].

Presentación y reflexión: Al final del día, todos los equipos se reúnen en la sede para ver lo que han hecho, y reflexionar colectivamente sobre los medios producidos (videos, fotografías, grabaciones, etc), sus respuestas a las preguntas, y las decisiones estratégicas tomadas durante el juego. Estos aspectos se hicieron aún más significativos para ellos cuando tuvieron que presentar sus resultados a una audiencia más amplia de los compañeros de clase. Por último, a cada grupo se le dice cuántos puntos ha ganado y el grupo con el mayor puntaje es anunciado [17].

Algunos inconvenientes en la práctica: Los alumnos no siempre fueron informados acerca de sus identidades como mendigo o comerciante. Esto significaba que se eliminó un cierto elemento competitivo del juego. Además, la tecnología no siempre funcionó según lo previsto. En particular en los primeros tres días de juego, se experimentaron problemas con el GPS que, o bien, mostró una posición incorrecta de los alumnos o no se mostraba en absoluto. El envío de fotografías y videos tomó un tiempo muy largo algunos días, lo que impidió a los alumnos la posibilidad de pasar a la siguiente tarea. No podían, por lo tanto, progresar en el juego y por lo tanto completaron menos sesiones de lo esperado [16].

3.5.2 HasleInteractive Mobile Urban Drama

Mobile Urban drama implementa una producción mediática (generación de medios por parte de los alumnos, por ejemplo videos, fotos, grabaciones) para apoyar experiencias atractivas a través de software para teléfonos móviles basados en localización [19].

La historia es un thriller ambiental que se lleva a cabo en parte en los teléfonos móviles de los alumnos y en parte en la naturaleza, y que debe ser investigada a través de ejercicios. El proyecto se llama HASLEINTERACTIVE y tiene lugar en un entorno natural de bosques, lagos y caminos. En el juego, los

usuarios están equipados con teléfonos móviles, auriculares y mapas. Interactúan con los diferentes escenarios de la obra a través de tecnología basada en la ubicación, como el GPS o con códigos de barras de las 2D. Además, reciben mensajes SMS y llamadas telefónicas, con ayudas o tareas que deberán ir resolviendo [19].

Síntesis del juego: Los alumnos han sido enviados en un viaje en el tiempo hasta el año 2022 cuando la naturaleza está infectada con un virus peligroso, que se extiende lentamente en el medio ambiente. Los alumnos son los protagonistas y actores activos en el drama, y se encuentran en un área que el virus todavía no ha infectado. Se les pide observar, describir e investigar diferentes biotopos con el fin de comprender cuando la naturaleza es sana y equilibrada. De esta manera, los alumnos van a ayudar a encontrar la clave para restaurar el equilibrio de la naturaleza. La historia se desarrolla en parte en los teléfonos móviles de los alumnos y en parte en el mundo real. Navegan a través de los mapas en sus teléfonos móviles. Además, reciben la ayuda investigadores científicos, guías de naturaleza e historiadores. La historia es contada por uno de ellos, que está en contacto con los alumnos a través de los teléfonos móviles (grabaciones MP3 en los teléfonos), lo que crea la ilusión de un contacto continuo. Sin embargo, debido a un “error técnico” en el equipo de comunicación de los alumnos, la señal desaparece de vez en cuando. En consecuencia, los alumnos también tienen que encontrar y escanear las etiquetas de códigos de barras 2D, que se colocan en la zona y después de haber explorado una etiqueta, ellos restablecen la señal, lo que motiva a una serie de diferentes estudios sobre el estado de la naturaleza. Los alumnos examinan el área, toman fotos y documentan en videos su trabajo [19].

En el camino, se encuentran con algunos códigos misteriosos, colocados en los árboles. Estos códigos deben ser decodificados, y los resultados se deben escribir en el teléfono móvil. Por último, a partir de los códigos recogidos, surge una frase real que revela una visión del estado de naturaleza [19].

En cada actividad los grupos se presentan con una misión que se divide en dos partes: en primer lugar a los alumnos se les pide que examinen la ubicación específica y busquen un ejemplo de animales, de plantas, de las condiciones de iluminación o de las condiciones del suelo. Esta primera parte tiene un tiempo limitado de cinco, diez o quince minutos y se sincroniza a través de los teléfonos del grupo. Durante este tiempo los alumnos reúnen materiales y toman fotos de sus hallazgos. En la segunda parte del trabajo, se le pide a los alumnos elegir papeles diferentes, bien como entrevistado o entrevistador, o camarógrafo, es decir, el entrevistador debe hacer las preguntas al entrevistado acerca de sus hallazgos, el entrevistado debe presentar los materiales, y el cámara graba la entrevista en video con el teléfono móvil [19].

Aspectos técnicos: A su llegada al escenario de juego, los alumnos reciben cada uno un teléfono móvil con el software y los mapas preinstalados y auriculares. Con respecto a los profesores, deben inscribirse en línea para un curso obligatorio

con el objetivo de hacer que puedan ayudar a los alumnos en el campo. Tienen además, la oportunidad de probar algunas de las tareas a las que los alumnos estarán expuestos, así como también recibir información práctica [19].

La duración del juego es de aproximadamente tres horas. Los materiales documentados son continuamente subidos a un servidor para que sean accesibles a los alumnos y profesores cuando regresen a la clase para finalizar el trabajo [19].

El sistema consiste en una aplicación móvil que se ejecuta en los teléfonos de los alumnos y controla el drama y las tareas. El teléfono móvil proporciona la interfaz para los elementos interactivos: etiquetas de exploración, sincronización, identificación de los símbolos en la zona, navegación, presentación de las tareas, producción de medios, etc [19].

La parte del servidor de la arquitectura proporciona dos servicios centrales: los servicios de sesión que se utilizan para manejar la dinámica de grupo y mantener la historia y sincronización a través de los teléfonos móviles de los alumnos y un servicio de comunicación que maneja el contenido producido durante los trabajos. Ambos servicios se implementan con el "framework urbanWeb", un framework basado en PHP con soporte específico para aplicaciones móviles, sensibles al contexto [19].

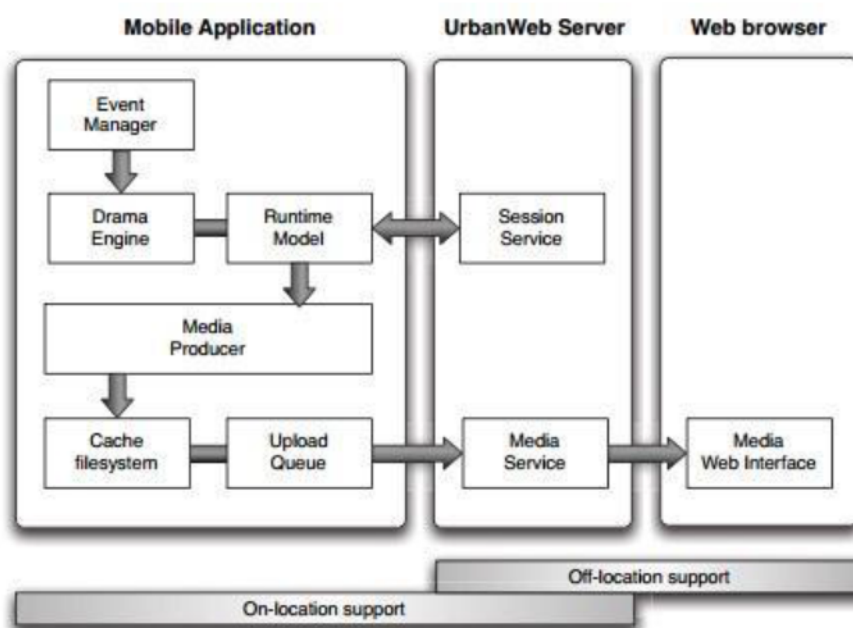


Figura 3. Arquitectura HasleInteractive. La aplicación y los servicios proveen soporte in situ, mientras que la interface web y los servicios "Urban Web Media" proveen el acceso a los materiales en la clase.

Otro aspecto importante del sistema es mantener un registro de los materiales producidos. Se proporciona una interfaz para subir desde el teléfono las grabaciones de fotos y videos. El profesor puede descargar todos los materiales producidos por una clase durante la excursión. Cuando los alumnos toman una foto o graban un video, el archivo se etiqueta con una marca de tiempo, la ubicación del

grupo y el número de grupo. Se almacena en el sistema de archivos móviles y se añade a una cola de subida. Un proceso en segundo plano supervisa la cola y carga los archivos al Servicio de Medios en el servidor. Todo esto sucede en background y los alumnos no tienen que esperar a que se suban sus grabaciones a la red pero puede continuar con el juego de inmediato. La cola se implementa con un mecanismo de persistencia en el teléfono, así que si algo sale mal durante la carga, la aplicación se puede iniciar en un momento posterior y terminar la carga de los materiales producidos [19].

3.5.3 Savannah

Se desarrolló y estudió un juego colaborativo basado en localización educativo llamado Savannah, en el que los niños aprenden acerca de la ecología de la sabana africana, especialmente sobre el comportamiento del león. Grupos de seis niños actúan en el rol de leones explorando una sabana virtual que está superpuesta sobre un campo de juego, una zona de césped abierto de aproximadamente el noventa por sesenta metros. Equipado con PDAs con Wi-Fi y GPS, los niños se mueven alrededor del campo de juego, explorando la variedad de terrenos de la sabana y descubriendo los recursos que los leones necesitan para sobrevivir. El juego implementa un enfoque sencillo basado en la ubicación en la que los jugadores deben estar en el mismo lugar con el fin de compartir información y actuar juntos [28].

Se divide en tres niveles, cada uno de los cuales duran aproximadamente ocho minutos. En el primer nivel, tienen que explorar y marcar el territorio. En el segundo y tercer nivel el deben “cazar” para alimentarse, primero en la temporada de lluvias abundantes y luego en la época de sequía más dura. Cada individuo necesita para maximizar su salud, hacerse de puntos de salud, los cuales se obtienen comiendo y bebiendo, se pueden perder por estar muerto de hambre, deshidratarse o ser atacados por otros animales. También es posible morir, en cuyo caso el león muerto tiene que volver al punto de partida física y sufrir una pena de varios minutos de tiempo antes de ser resucitado y enviado de vuelta a la sabana. Los puntos se otorgan de acuerdo con la salud de cada león al final de cada nivel [28].

El juego requiere de la colaboración, al igual que una manada de leones tiene que trabajar juntos para sobrevivir, por lo que los jugadores tienen que trabajar juntos para ganar el partido, decidir qué animales atacar y cuántos leones tienen que atacar en conjunto con el fin de tener éxito. En concreto, los jugadores se mueven alrededor del campo con la presentación de informes de sus posiciones GPS que son transformadas a sus posiciones en la sabana virtual. Cada vez que un jugador entra en un nuevo escenario, recibe una nueva combinación de sonido e imagen y, posiblemente, un botón de acción si contiene un objetivo activo [28].

Cada segundo, el PDA de un jugador transmite su localización a un servidor central de juego a través de WiFi. El servidor responde mediante la activación de la

presentación de sonido (con auriculares), imágenes estáticas y la información de estado en la PDA, revelando la sabana virtual y sus habitantes. Un jugador invoca una acción como un ataque al tocar un botón en la pantalla táctil de la PDA, donde la acción apropiada se determina por la ubicación actual y el nivel de juego del jugador. Una notificación de esta acción se envía al servidor de juegos a través de WiFi, que responde con una notificación de éxito o fracaso. El PDA desactiva más acciones del usuario a la espera de esta respuesta desde el servidor de juego. Sólo puede haber un ataque en curso en un escenario dado en cualquier momento. Cuando un jugador ataca, por consiguiente, inicia ya sea un nuevo ataque o se une en un proceso continuo. Cuando se inicia un nuevo ataque, los demás jugadores tienen diez segundos para unirse antes de que el ataque haya sido resuelto [28].

Algunas dificultades técnicas: Mientras que los jugadores mostraron un enfoque fluido en cuanto a la formación de grupos, el sistema incorporó una visión más rígida, lo que llevó a dificultades con el intercambio de contexto y la coordinación de acciones, sobre todo cuando los grupos de jugadores atraviesan fronteras en escenarios virtuales o inician acciones mientras están en movimiento. Los ataques de coordinación se volvieron difíciles de coordinar y frustrantes a veces. El problema más evidente fue el de establecer un contexto compartido (a veces jugadores que estaban junto a otros, deliberadamente tratando de coordinar sus acciones, no podían adquirir la misma información). La comprensión de estas dificultades de coordinación requiere descomprimir la compleja relación entre el comportamiento de los jugadores y las características de la tecnología [28].

4. Modelo de Juego Colaborativo Móvil

El tipo de juego que se quiere modelar involucra aspectos relacionados por lado al posicionamiento de los usuarios y por otro la colaboración entre los mismos.

El posicionamiento involucra cuestiones relacionadas a la movilidad del usuario, por lo tanto es necesario modelarlo. Tengamos en cuenta que este tipo de aplicaciones consideran a la posición como un aspecto relevante, asumen que el alumno o jugador se mueve continuamente de una posición a otra, obteniendo así, un nuevo contenido acorde a su posición [30].

Es importante también tener en cuenta el rol comunicacional del juego, al ser un juego colaborativo, se necesita un soporte íntegro para la interacción entre las partes involucradas. En nuestro caso, por ejemplo los subgrupos, como veremos más adelante, deben conocerse y poder comunicarse de forma fluida sin que esto represente un problema a nivel de modelo.

A continuación explicaremos las características del tipo de juego que se quiere modelar junto con el modelo propuesto y explicar cómo son las interacciones entre las clases más importantes.

4.1 Modelo Propuesto

Un **juego** se define como un conjunto de consignas que posteriormente serán asignadas a los grupos que juegan. No necesariamente todas las consignas serán asignadas. Por ejemplo, en el caso de un juego que consiste en tipificar ciertos animales, dos consignas de juego podrían ser: “Hallar animales mamíferos” y “Hallar animales herbívoros”.

Todo lugar físico destacable en el juego se representa como un punto de interés (en adelante **POI**), el cual tiene la posición real.

Cada **consigna** tendrá un conjunto de piezas de interés, o **piezas a recolectar** asociado. Cada pieza posee cierta información, y tiene asociado un POI, el cual determina la posición donde estará dicha pieza.

El **grupo** de juego estará formado por uno o más subgrupos. Cada **subgrupo** tendrá asignado al menos un **participante** y tendrá en cuenta la consigna del grupo al que pertenece. Cada subgrupo cuenta (físicamente) con un dispositivo móvil desde el cual interactúa con el juego en cuestión.

Cada subgrupo puede tener, asignado un **dispositivo** para jugar. Esto permite contemplar características particulares de cada celular, como por ejemplo, si

dispone de bluetooth o cámara fotográfica. Esta asignación se considerará cuando las características del dispositivo afecten de alguna manera el desarrollo del juego.

Un juego tiene un **punto de encuentro inicial**, que tiene asociado el lugar físico (POI) en donde los grupos se encuentran para comenzar a jugar y un **punto de encuentro final**, lugar en el que los grupos se deben encontrar al finalizar el juego. La posición del punto de inicio y punto de fin pertenecen al espacio de juego, y está determinada por el POI que tienen asociado.

El juego posee una **estrategia**, que podría consistir en mostrarle a cada grupo solamente los POIs que debe visitar (estrategia de *Buscar Elementos*). Otra estrategia podría consistir en mostrarle a cada subgrupo los POIs del juego que deben visitar y que al visitarlos determinen si cumplen o no con la consigna que tienen asignada (estrategia de *Descubrir Elementos*).

Al comenzar el juego, cada subgrupo visualizará, por ejemplo, en un mapa el punto inicial del juego y una vez que todos los grupos llegan a dicho punto, cada uno de los mismos recibirá la consigna a cumplir y visualizarán en el mapa los POIs del juego asociados a sus piezas dependiendo de la estrategia del mismo.

Veamos cual es la dinámica de la estrategia de *Descubrir Elementos*.

Cuando un subgrupo llega a un POI, debe decidir, si la pieza asociada al POI cumple o no con la consigna que se le asignó al grupo al que pertenece. En este momento, el subgrupo en cuestión, tiene dos alternativas:

- Consultar por única vez a los otros subgrupos que conforman su grupo, para ello, se genera una consulta con una decisión propuesta. Cuando se realiza la consulta, se propone una decisión en la que se explicita si a consideración del subgrupo que consulta, la pieza recolectada en el POI cumple o no con la consigna y una justificación de su propuesta. Entonces cada subgrupo recibe la consulta generada por el subgrupo y genera una respuesta a la consulta en la que dice sí a su entender la propuesta es o no válida y da su propia justificación a su propuesta. Una vez que el subgrupo que consultó recibe las respuestas, las evalúa y toma una decisión final, la cual se conoce como la decisión tomada.
- Tomar una decisión final (decisión tomada) sin realizar una consulta al resto de los subgrupos.

Cuando un grupo termina de visitar todas las piezas a recolectar, éste le mostrará, por ejemplo, en un mapa, el punto de encuentro final.

Cuando todos los grupos llegan al punto de encuentro final, el juego finaliza y

se muestran los resultados de todos los grupos y para el subgrupo en particular, se informa cuál fue el resultado de cada decisión tomada sobre las piezas que visitó.

Considerando las características descritas, en la Figura 4 se presenta el modelo propuesto

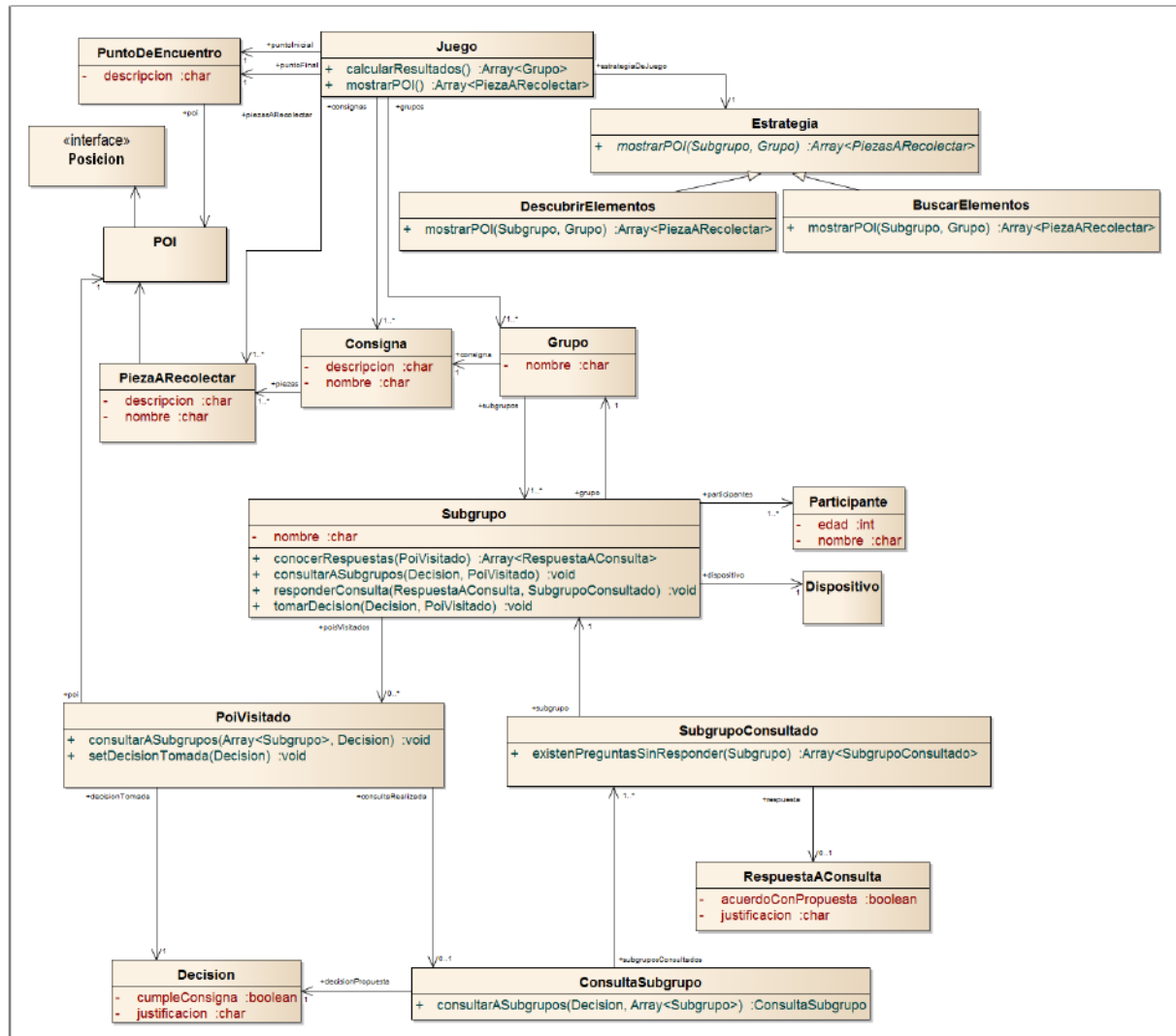


Figura 4: Modelo propuesto para Juegos Colaborativos Móviles

4.2 Caracterización General del Modelo

En esta sección se analizará en detalle las decisiones generales tomadas en el modelo propuesto, respecto a las estrategias de juego, los grupos, consignas, piezas a recolectar.

- El juego y su estrategia

Comencemos por el juego y la estrategia del mismo, podemos observar como el juego (clase *Juego*) conoce a su estrategia (clase *Estrategia* a través de la variable de instancia *estrategiaDeJuego*), como se puede apreciar en la Figura 5. Como vimos pueden existir diferentes estrategias (nombramos dos, *DescubrirElementos* y *BuscarElementos*), acorde a las características del juego que se quiera representar. Dependerá también de diversos factores, por ejemplo, el ámbito del juego, la orientación hacia lo colaborativo, los recursos disponibles, etc. Cada instancia de juego tendrá asociada una estrategia en particular que no va a variar durante el juego.

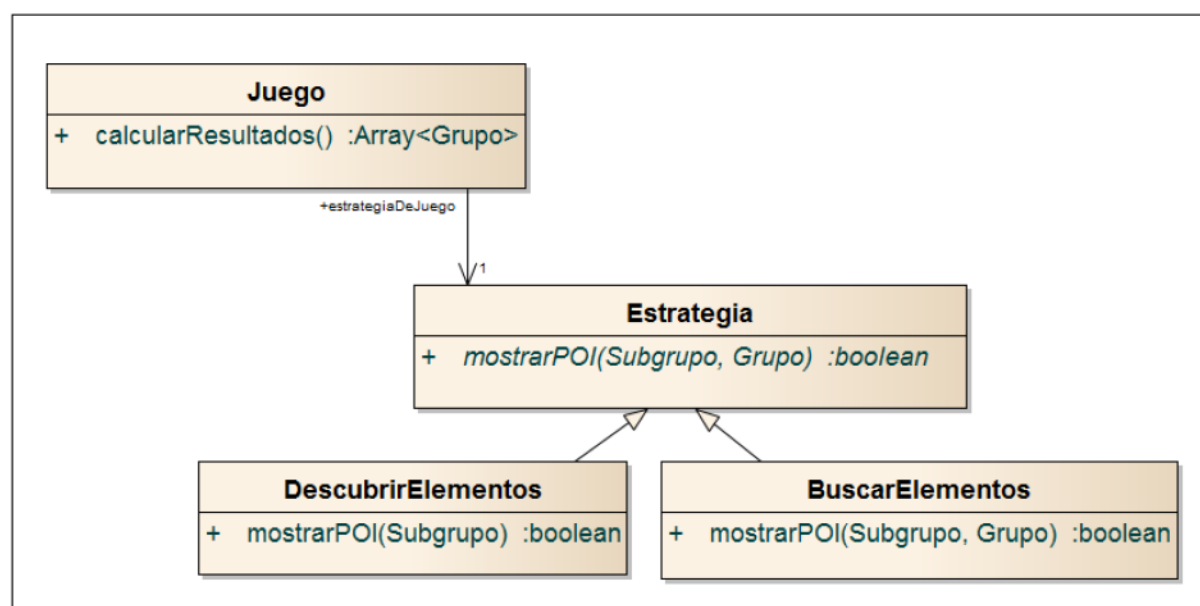


Figura 5: Juego y su Estrategia

- **El punto de encuentro inicial y final del juego**

La clase *Juego* conoce además de su estrategia, sus puntos de encuentro inicial y final (clase *PuntoDeEncuentro* a través de sus variables de instancia *puntoInicial* y *puntoFinal*) que necesitan tener asociada una ubicación precisa (clase *POI* con su posición) para poder indicar donde comienza y finaliza físicamente el juego. Esto se puede observar en la Figura 6. Cada subgrupo visualizará los puntos de encuentro inicial y final, y tendrá que acercarse físicamente a ellos, cuando lo estén¹, el juego continuará con su dinámica.

Los puntos de interés (POI) se modelan como una clase particular para representar todo punto físico destacable en el juego. Es decir, todo lo que debe estar posicionado conoce a una instancia de esta clase, como es el caso de los puntos de encuentro.

¹ La posición de los subgrupos se puede detectar acorde a algún mecanismo de sensado, por ejemplo, porque los mismos leen un código QR o acorde a los datos GPS.

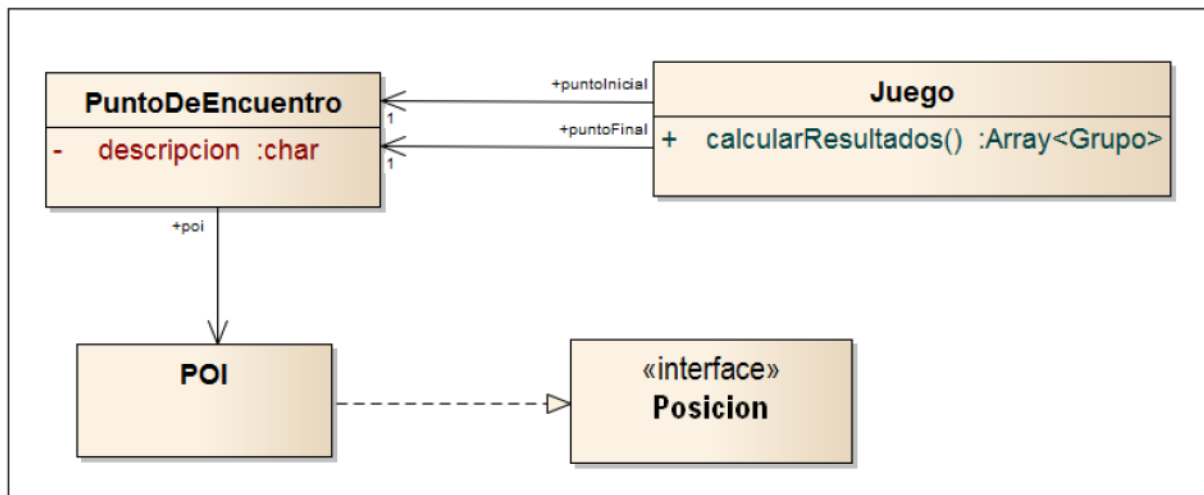


Figura 6: El punto inicial y final de encuentro

- **Las consignas, sus piezas a recolectar y POIs**

La clase *Juego*, ésta posee también conocimiento de todas las piezas a recolectar existentes en el espacio de juego (clase *PiezaARcolectar* a través de su variable de instancia *piezasARcolectar*), independientemente de las consignas que tengan asignadas. Como se puede apreciar en la Figura 6. Esta información será usada en el caso de necesitar mostrar todas las posibles piezas a recolectar en un lugar determinado.

Para concluir con la clase *Juego* veamos uno de sus puntos claves, sus consignas (clase *Consigna*), cada juego puede tener asociadas diferentes consignas (1 o varias), el modelo permite que estas puedan ir variando de acuerdo al objetivo educacional del juego, es decir, por ejemplo las que vimos (animales mamíferos y animales) pueden emplearse en materias afines sobre ciencias naturales, pero perfectamente se pueden crear consignas referidas a historia, geografía, etc. Esta asociación permite que el juego (clase *Juego*) pueda calcular el resultado final, basándose en la estrategia del mismo, sus consignas y a través de estas, la efectividad de los grupos con respecto a la visita y las decisiones (si tuviesen que tomar alguna) sobre sus piezas a recolectar.

La consigna (clase *Consigna*) tiene asociada las piezas a recolectar que cumplen con su descripción, de este modo a la hora de evaluar los resultados, una pieza es válida para un grupo (clase *Grupo*) si la consigna asociada al grupo (variable de instancia *consigna*) posee la pieza asociada (está entre su colección de piezas). Recordemos que cada pieza a recolectar tiene un punto de interés (clase *POI*) con su posición específica para poder ser visitadas².

² Para detectar que un grupo o subgrupo visita una pieza habrá un mecanismo de sensado verificando cuando se

Las clases antes descritas se pueden visualizar en la Figura 7.

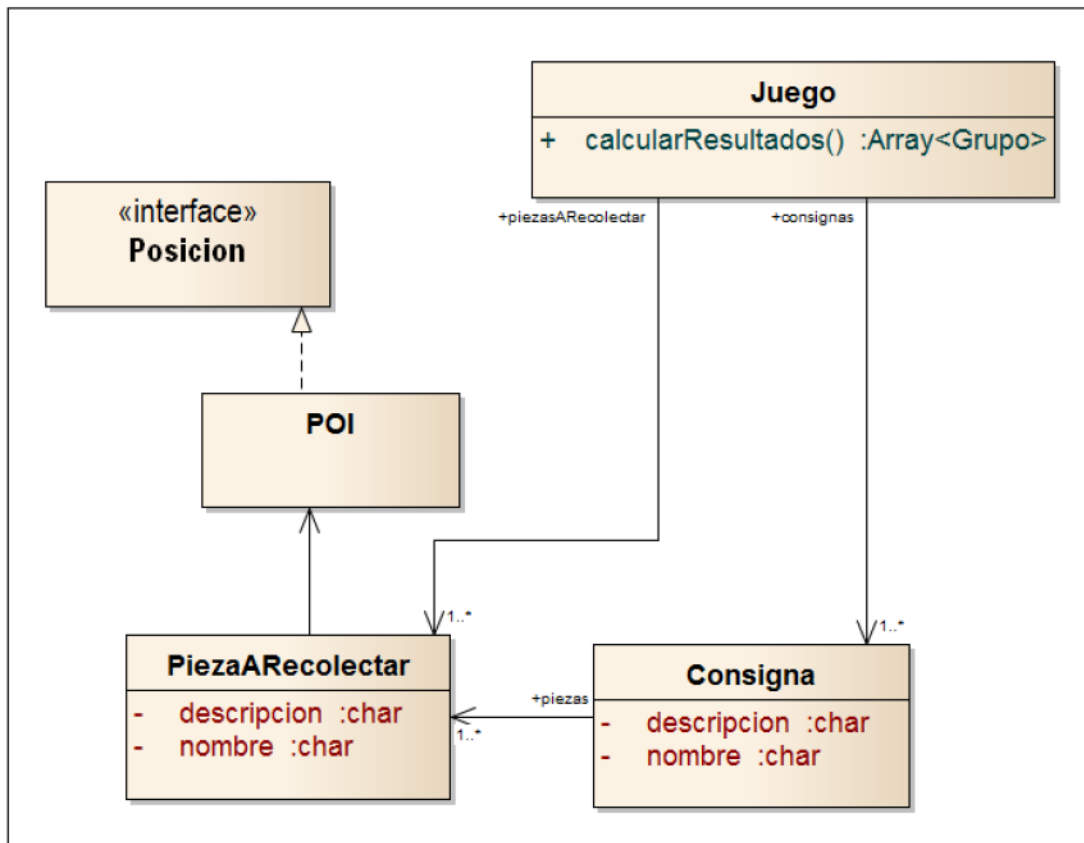


Figura 7: Consigna y Pieza a Recolectar

En las Figuras 8 y 9 se muestran dos modelos de instancias, en la primera se muestra un modelo acotado donde una pieza cumple con la consigna, y en la otra figura la pieza no cumple la consigna. Como vemos en los diagramas de instancia, la clase Geométrica, implementa la interfaz Posición.

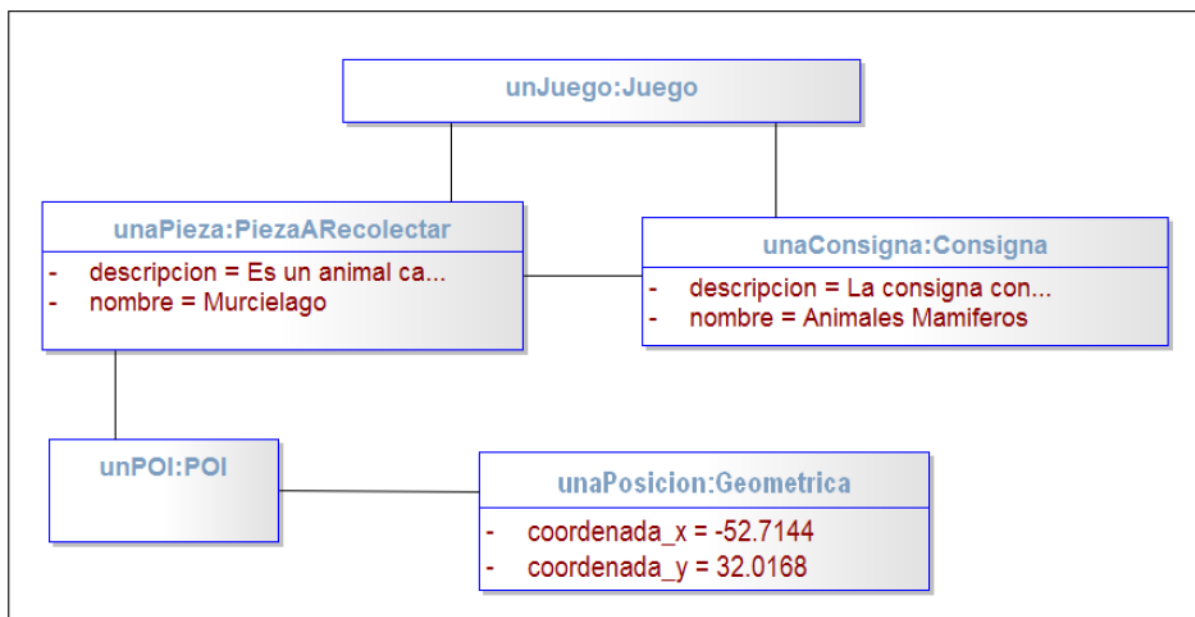


Figura 8: Modelo de instancia en el que la pieza cumple la consigna

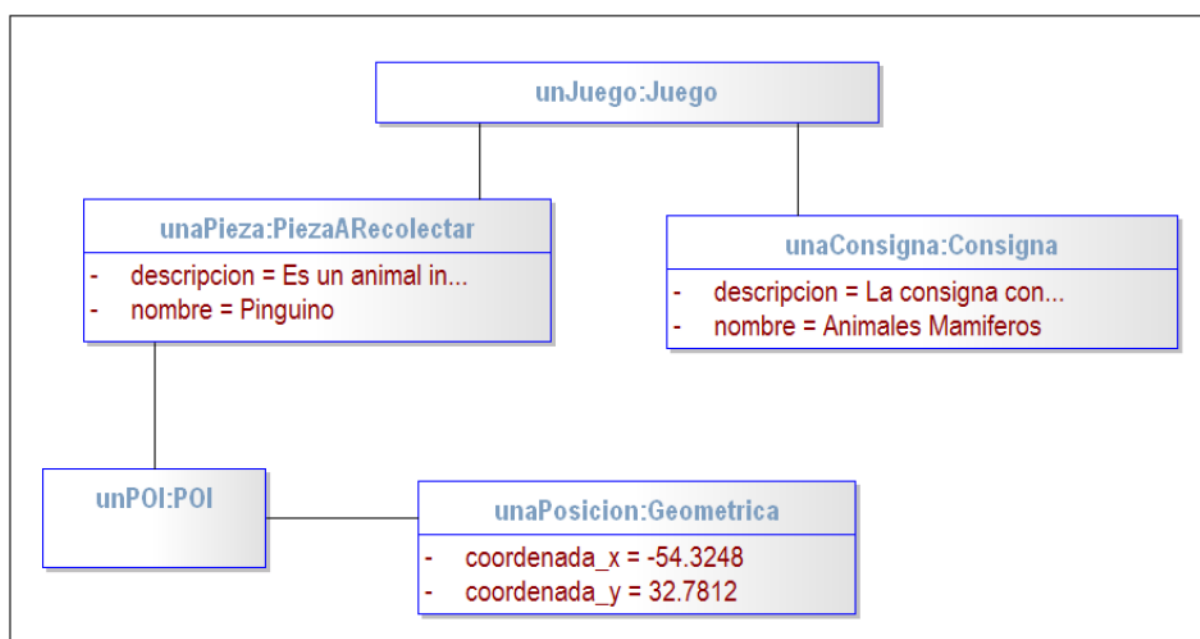


Figura 9: Modelo de instancia en el que la pieza no cumple la consigna

- **El grupo, su consigna, subgrupos y participantes**

Continuemos ahora con el grupo (clase *Grupo*) y sus aspectos relevantes como puede apreciarse en la Figura 10. El grupo como dijimos tiene, asignada a través del juego, una consigna (*Consigna* a través de su variable de instancia *consigna*) y uno o varios subgrupos (clase *Subgrupo* a través de *subgrupos*) asociados, para que resulte el componente colaborativo, cada grupo debería tener al menos dos subgrupos asociados (si un grupo consiste sólo de un subgrupo, se pierde la propiedad colaborativa del juego). Los subgrupos tendrán siempre presente la consigna del grupo (que conoce a la vez por su variable de instancia

grupo) a la hora de visitar las piezas y tomar las decisiones. Por último, cada subgrupo tiene un nombre que lo identifica (y puede tener un password, que servirán de ayuda a la hora de loguear su dispositivo a la aplicación) y uno o varios participantes (clase *Participante*), de los que se conoce su nombre y edad.

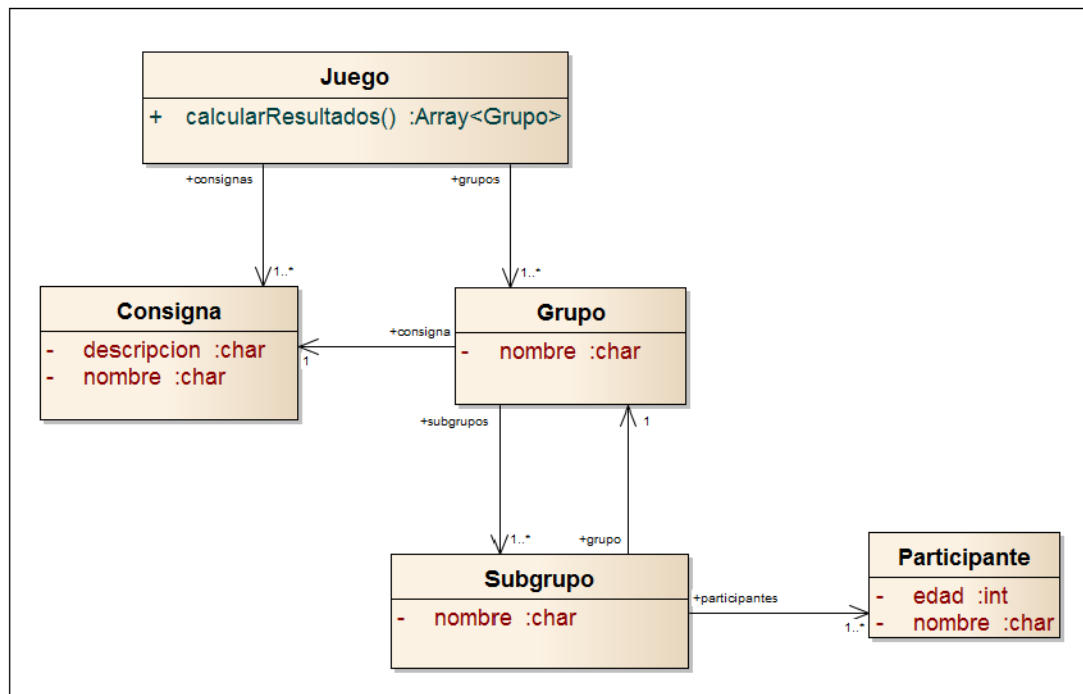


Figura 10: Modelado de los Grupos

• Subgrupo y Dispositivo

Como se puede apreciar en la Figura 11, se modela el dispositivo que posee cada subgrupo para realizar el juego. A nivel de modelo, se agrega esta clase para representar que el dispositivo podría ser relevante para el juego, cuando el mismo reacciona acorde a alguna característica de dicho dispositivo. Esto es algo que se deja abierto a nivel de modelo, como un futuro punto de extensión, y que no se explorará en esta tesis.

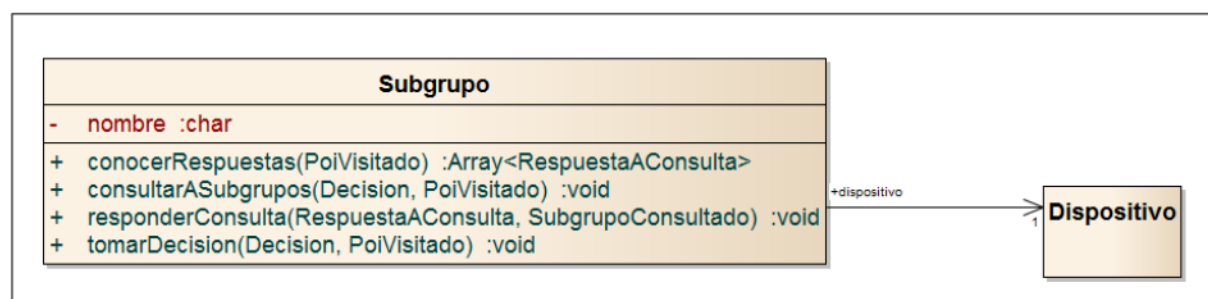


Figura 11: Dispositivo del Subgrupo

4.3 Aspectos Colaborativos del Modelo

Centrémonos ahora en una de los aspectos más interesantes de la tesis y del modelo en cuestión, lo colaborativo.

Como dijimos, una vez que comienza el juego, cada subgrupo visualizará el punto de encuentro inicial, se acercará a éste y una vez que todos los subgrupos se acerquen al mismo, a cada subgrupo se le desplegarán en el mapa las piezas a visitar. Entonces la tarea del subgrupo es visitar esa pieza, una vez que llegan se les desplegará información sobre la misma y deberán tomar una decisión.

- **Las consultas**

Cuando llega a una pieza determinada, se agrega el poi de la pieza a la colección de pois visitados del subgrupo (variable de instancia *poisVisitados*). A la hora de decidir, el subgrupo deberá considerar o bien, consultar a los demás subgrupos (conocidos a través del conocimiento del grupo) o, si está seguro de la respuesta, tomar una decisión final sobre la pieza en cuestión. En la Figura 12 se puede apreciar las principales clases involucradas en las consultas y la decisión de los subgrupos.

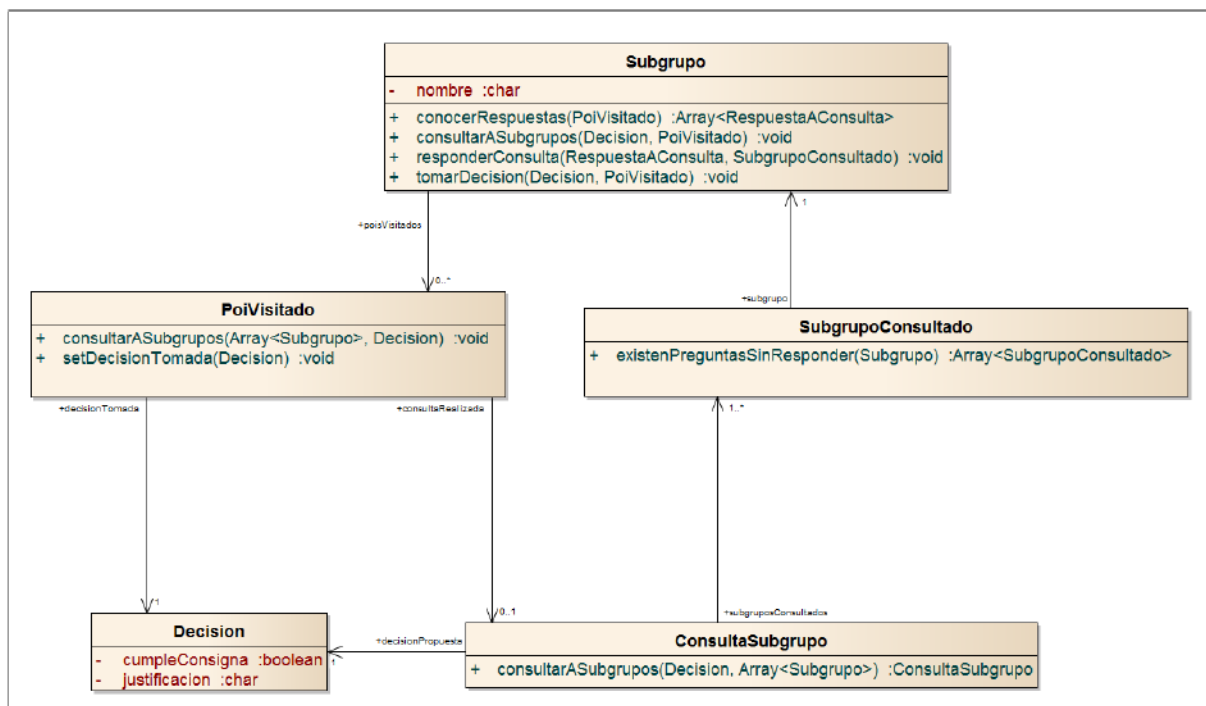


Figura 12. Las clases involucradas en las consultas

Veamos cómo funciona la consulta a los subgrupos conocidos en el modelo. A través del método *consultarASubgrupos(PoiVisitado,Decision)*, el subgrupo será

capaz de enviar una decisión parcial (llamada *decisionPropuesta*, que es variable de instancia de la clase *ConsultaSubgrupo*), y esta decisión parcial será usada como ayuda por cada subgrupo que deba responder. Es decir, primero el subgrupo crea una decisión parcial (clase *Decision*), con una justificación parcial, y una variable de condición booleana en la que especifica si cumple o no la consigna propuesta (ambas variables de instancia de la clase *Decision*).

Cuando se invoca al método *consultarASubgrupos* de la clase *Subgrupo* enviando como parámetros, el POI visitado con la pieza en cuestión y la decisión parcial, este delega a la clase *POIVisitado* la tarea enviándole además como parámetro los subgrupos que deben responder (conocidos por el subgrupo a través del grupo en común). La clase *POIVisitado* a su vez vuelve a delegar la tarea a la clase *ConsultaSubgrupo* enviándoles los parámetros indicados en la figura. Por último la clase *ConsultaSubgrupo* crea un objeto de la clase *SubgrupoConsultado* por cada subgrupo de la colección enviada como parámetro, y a cada uno le asocia el subgrupo que consulta. Luego lo agrega a su colección de *subgruposConsultados* y agrega la decisión parcial enviada como parámetro en su variable de instancia *decisionPropuesta*. Esta serie de delegaciones la podemos apreciar mejor en el diagrama de secuencia que se muestra la Figura 13.

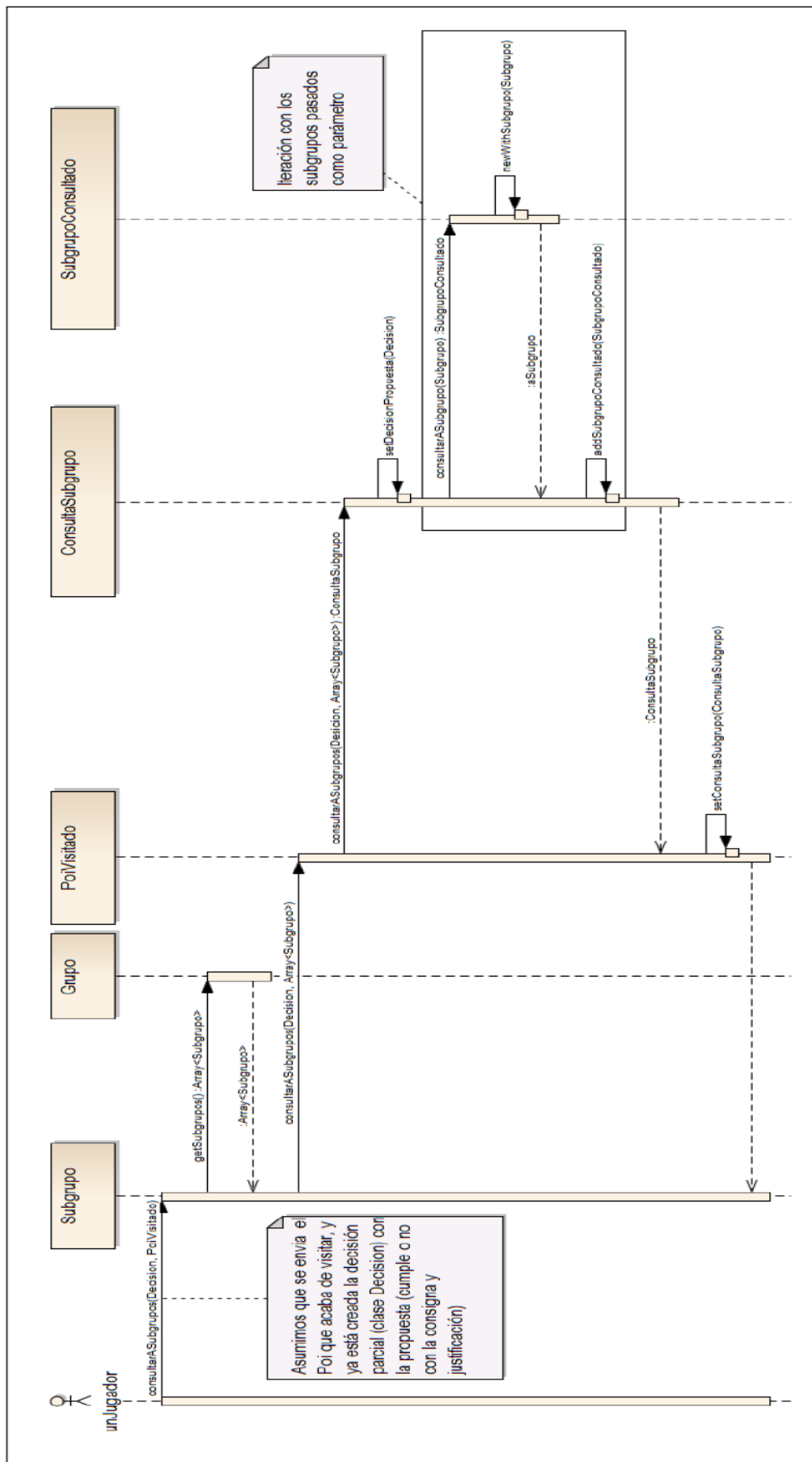


Figura 13. Diagrama de secuencia del método consultarASubgrupos()

- **Las respuestas**

Si un subgrupo se dispone a responder una consulta realizada en pos de colaborar y ayudar a completar la tarea a su subgrupo conocido, debe responder basándose en el conocimiento y experiencia de los participantes del subgrupo, o simplemente avisar que no puede ayudar. A través del alcance del modelo, dispone de la información (nombre y descripción) de la consigna (clase *Consigna*) en común, la información de la pieza a recolectar (nombre y descripción) del subgrupo que consulta (clase *PiezaARecolectar*) y de la información (si cumple o no con la consigna y la justificación propuesta) de la decisión parcial (Clase *Decision*) que tomó el subgrupo que necesita ayuda. Deberá entonces especificar si está de acuerdo o no con la decisión del subgrupo que consulta y justificar su respuesta. Para este propósito existe un método en la clase *subgrupo* llamado *responderConsulta()* que simplemente debe crear un objeto *RespuestaAConsulta*, asociar a sus variables de instancia (acuerdoConPropuesta y justificación) la información generada y finalmente asociarlo al objeto de la clase *SubgrupoConsultado* que fue devuelto en el llamado del método *existenPreguntasSinResponder()* visto en la sección anterior. Las clases mencionadas se pueden apreciar en la Figura 14.

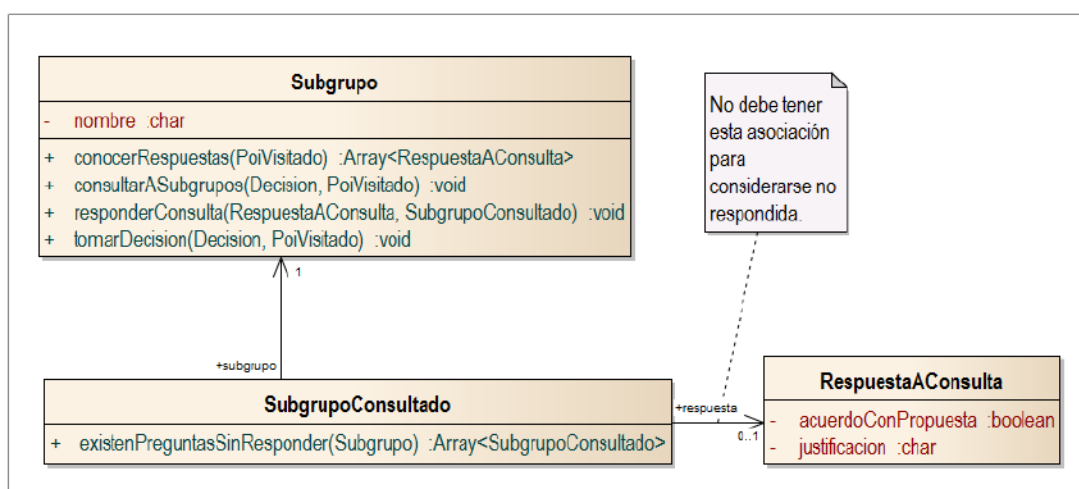


Figura 14. Las clases involucradas en la respuesta

El modelo está pensado para soportar que un subgrupo no responda a una consulta realizada. Esto se representa con la clase *SubgrupoConsultado* que tienen al subgrupo como variable de instancia y no tiene asociada ninguna respuesta (clase *RespuestaConsulta*, variable de instancia de la clase *SubgrupoConsultado*). Como se pudo observar en la Figura 14.

- **Vislumbrando las respuestas**

Una vez que un subgrupo realiza una consulta sobre una pieza a recolectar, como dijimos, se queda esperando a que sus compañeros de los otros subgrupos respondan, es por esto que en modelo, en la clase *Subgrupo* existe el método *conocerRespuestas()*, que dado un *PoiVisitado* (el asociado a la pieza consultada), busca que objetos *SubgrupoConsultado* tienen asociada una respuesta (clase *RespuestaAConsulta*) y los devuelve en una colección. Veamos en la Figura 15 un diagrama de secuencia cómo funciona el método.

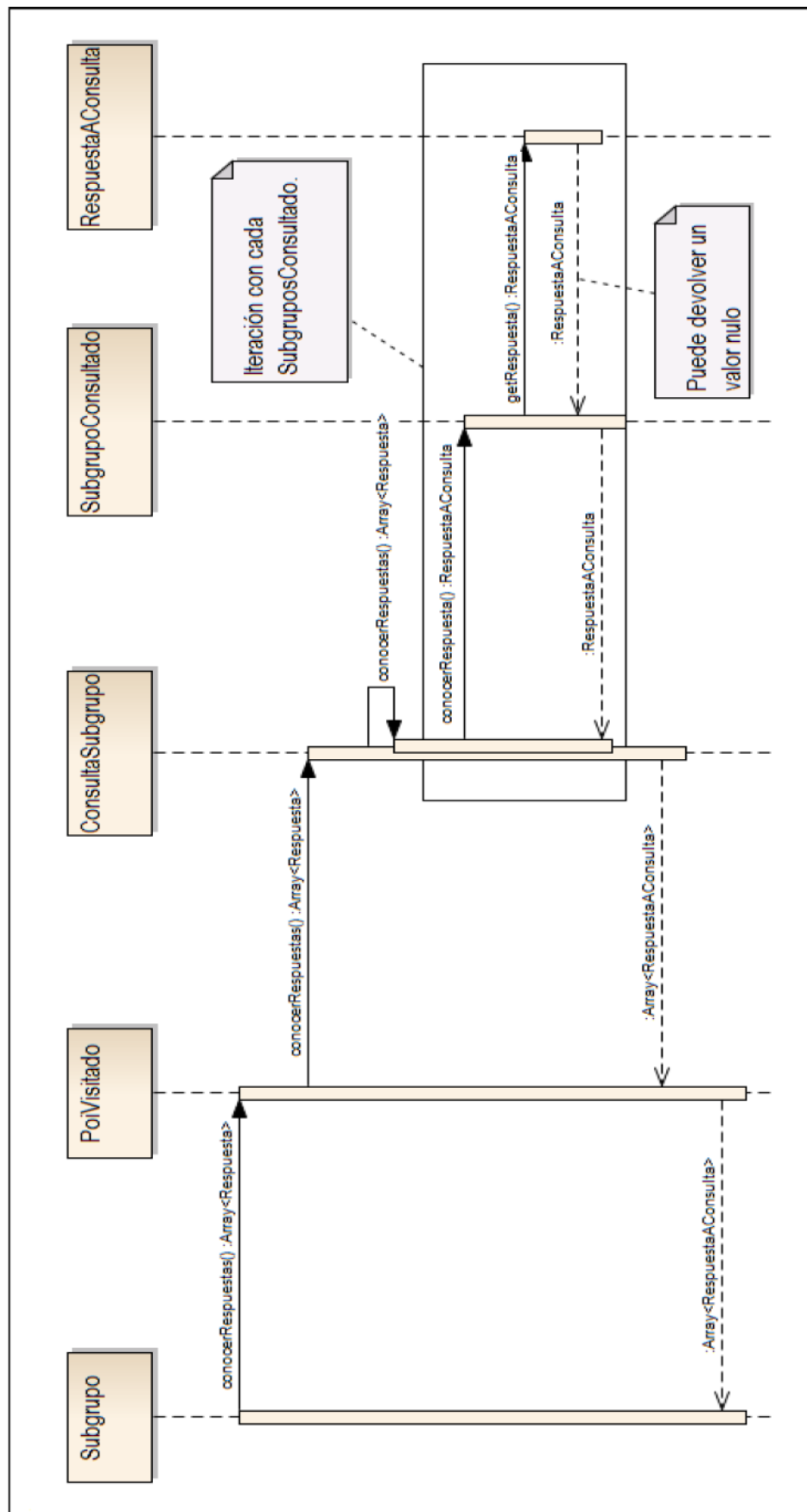


Figura 15. Diagrama de secuencia del método conocerRespuestas()

- **La decisión final**

Finalmente en un momento dado, ya sea porque no necesita consultar o bien realizó una consulta y teniendo en cuenta la ayuda de los subgrupos compañeros, un subgrupo toma o debe tomar una decisión final sobre una pieza a recolectar. En la clase *Subgrupo* existe un método para tal fin llamado *tomarDecision()*, al que se le envía como parámetro el poi visitado que hace referencia a la pieza y se le envía la decisión final, que posee información sobre si la pieza cumple o no con la consigna y una justificación. Las clases involucradas se pueden apreciar en la Figura 16.

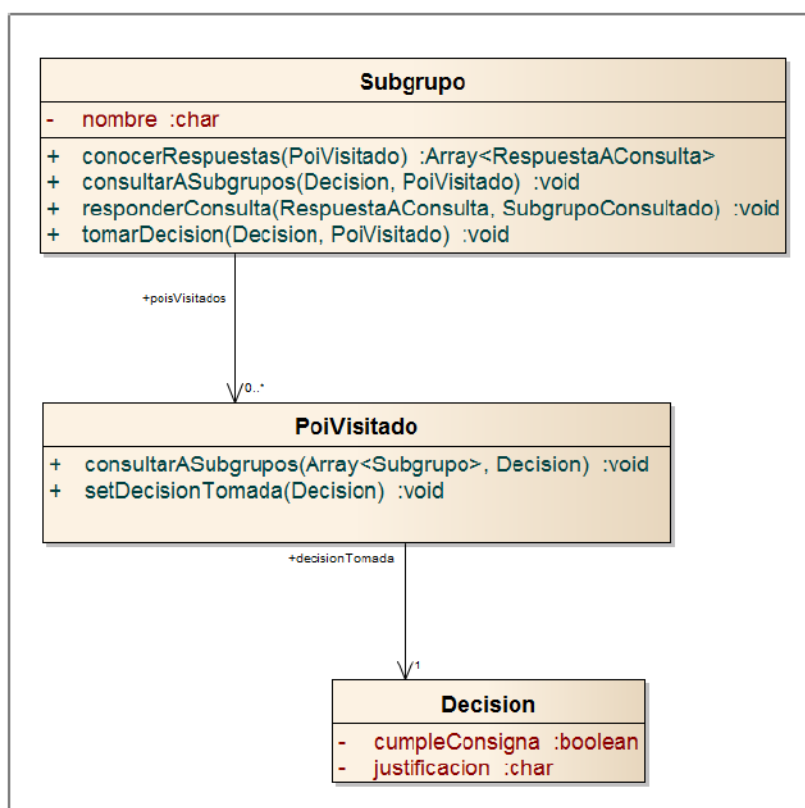


Figura 16. Las clases involucradas en la decisión final.

5. Implementación del Prototipo

En el Capítulo 4 hemos descrito en profundidad el modelo propuesto para Juegos Colaborativos Móviles, ahora nos centraremos en la implementación del prototipo que realizamos para esta tesina. Nos hemos propuesto realizar un desarrollo con los paradigmas aprendidos durante nuestra formación académica teniendo siempre como premisa utilizar buenas prácticas de programación para posibilitar, si es necesario en un futuro, que el prototipo sea escalable sin mayor dificultad. Haremos una descripción de la arquitectura elegida teniendo en cuenta lo dicho anteriormente y además, el modelo y la dinámica del juego. Finalmente mostraremos una simulación de un ejemplo del juego.

5.1 Arquitectura

Nuestra aplicación se divide básicamente en dos partes, por un lado una aplicación nativa que se instala y ejecuta en dispositivos móviles y por otro lado una aplicación web que se instala y queda accesible en un servidor web. Asociado al servidor web existe un servidor de base de datos que persiste el modelo de datos del juego.

En el servidor está la representación del modelo del juego, con los grupos, subgrupos, piezas, etc., es el encargado de mantener una coherencia en el desarrollo del juego al conocer los estados de los distintos subgrupos, permitiendo visualizar diferente información en cada dispositivo o bien sincronizando la llegada de los subgrupos a un punto en común.

Mientras que del lado del cliente se posee una aplicación nativa, en la que se despliega y se persiste información del subgrupo asociado al dispositivo, tal como su posición actual, su estado actual, que piezas visitó, qué decisión tomó en cada una de ellas, cuales quedan por visitar, etc.

En la Figura 17 se puede apreciar la Arquitectura del prototipo.

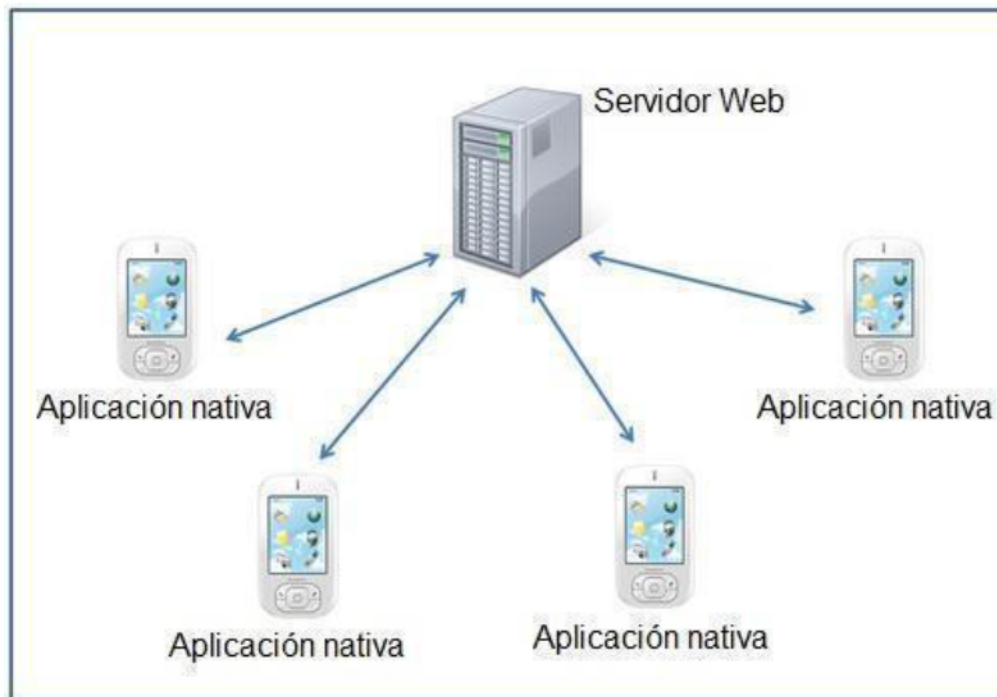


Figura 17. Arquitectura del Prototipo

Como se pudo observar en la Figura 17 los equipos clientes, que forman parte de una red, se comunican con un servidor, un equipo generalmente potente en materia de capacidad de entrada/salida, que proporciona servicios a los mismos. Estos servicios son utilizados por programas denominados programas clientes, que se ejecutan en equipos clientes. Las principales ventajas de la utilización de una arquitectura como la propuesta son, entre otras, **recursos centralizados** debido a que el servidor es el centro de la red, puede administrar los recursos que son comunes a todos los usuarios, por ejemplo: una base de datos centralizada. **Red escalable**, gracias a esta arquitectura, es posible quitar o agregar clientes sin afectar el funcionamiento de la red y sin la necesidad de realizar mayores modificaciones [31].

- **Características de la Aplicación Web (Servidor)**

Para la implementación del prototipo, decidimos utilizar un servidor HTTP Apache 2.4.7 [32], MySQL 5.5 [33] como motor de base de datos, PHP en su versión 5.4 [34] como lenguaje de programación del lado del servidor.

La aplicación web para la configuración y administración del juego la realizamos con el framework Symfony en su versión 1.4 [36] buscando optimizar el desarrollo de la misma, ya que brinda herramientas y clases enfocadas en reducir el tiempo de desarrollo, automatizando las tareas simples y permitiéndonos dedicarnos por completo a los aspectos específicos del juego. Le adaptamos una plantilla de estilos [38] la cual nos brindó la posibilidad de generar una interfaz más amigable al usuario final que la que viene por defecto en esta clase de frameworks.

- **Características de la Aplicación Nativa (Clientes)**

Por el lado del cliente nos propusimos el desafío de desarrollarlo para que corra bajo el sistema operativo Android [35], lo cual implicó una curva de aprendizaje algo elevada ya que no teníamos una fuerte experiencia previa con respecto al lenguaje de programación utilizado. Aunque creemos que el esfuerzo finalmente valió la pena al lograr un prototipo funcional y estable en los tiempos previstos.

Android se desarrolla bajo el lenguaje de programación Java [44], utilizando también un conjunto de herramientas de desarrollo (Android SDK, Software Development Kit de Android [39]), que nos provee de un depurador de código, bibliotecas, un simulador de dispositivos, documentación, ejemplos de código y tutoriales.

- **La comunicación**

Como vemos en la Figura 18, el método de comunicación del prototipo es a través de solicitudes y respuestas. Los clientes realizan las solicitudes, y esperan la respuesta del servidor. En la implementación, además de la comunicación asincrónica, en la que los dispositivos (subgrupos) envían o solicitan determinada información y siguen el curso normal de ejecución, nos vimos en la necesidad de realizar una comunicación sincrónica, por ejemplo barreras, en los casos en los que los subgrupos se quedan esperando a que los demás subgrupos lleguen a un determinado lugar o estado y es el servidor quien debe avisar que pueden continuar. También de acuerdo al desarrollo del juego, se dan casos en los que los subgrupos se comunican entre ellos, se consultan y colaboran, en estos casos en el prototipo (de manera transparente al jugador), el servidor hace de mediador y “avisa” a los dispositivos involucrados la llegada de nuevos mensajes o la necesidad de una respuesta ante una consulta.



Figura 18. Comunicación Cliente/Servidor

Las razones de utilizar esta arquitectura son varias. Primero, la necesidad de representar y persistir los datos comunes a todos los subgrupos en un lugar común (servidor de base de datos), tales como las piezas del juego, los grupos involucrados, las consignas, etc. Cada cliente a su vez, guarda información sobre su estado, aunque ésta es volátil (si el dispositivo falla o se apaga se puede perder la información) y es válida sólo para el subgrupo asociado al dispositivo. Finalmente, es el servidor en definitiva quien mantiene información sobre todos los subgrupos

involucrados, facilita la comunicación entre ellos y mantiene la coherencia del juego.

A la hora de implementar la comunicación entre el cliente y el servidor se nos presentó un desafío particular, no nos fue fácil establecer una comunicación simple y estable, entre el servidor, utilizando PHP bajo el framework Symfony y Java en los clientes Android. Decidimos por tal motivo utilizar Web Services [48] bajo el protocolo SOAP (Simple Object Access Protocol [47]), para estandarizar los llamados y las respuestas de un lado y del otro.

Por el lado de Symfony (en el Servidor), utilizamos un plugin llamado ckWebServicePlugin [45] que nos permitió construir Web Services [48]. A partir de la devolución de objetos simples o complejos según la necesidad del juego, genera un archivo WSDL[51] que facilita la interoperabilidad con diferentes clientes, Java por ejemplo. La complejidad del plugin varía dependiendo de los parámetros que recibe el método y la respuesta que devuelve, no es lo mismo retornar un objeto simple (integer, string) que un objeto complejo como una colección de piezas a recolectar.

En la Figura 19 se puede apreciar como el plugin utiliza los comentarios de los métodos para crear el archivo wdsi. En este caso en los comentarios del método getPiezas(), se especifica el nombre del web service 'WSJuegoColaborativo', el nombre del método 'getPiezas', el parámetro que se debe enviar identificando al subgrupo, y finalmente, lo que retornará en método, una colección de la clase Pieza.

```
/*
 * Devuelve las piezas a recolectar asociadas a un subgrupo
 * @WSMethod(name='getPiezas', webservice='WSJuegoColaborativo')
 * @param integer $idSubgrupo Identificador del subgrupo
 * @return Pieza[]
 */
public function executeGetPiezas($request) {
    ...
}
```

Figura 19. Descripción del método getPiezas()

En el cliente (Aplicación Android) lo resolvimos de la siguiente manera, cuando un objeto necesita comunicarse con el servidor, crea una instancia de una clase (WSTask) que extiende de la clase AsyncTask [49], ésta proporciona la posibilidad de ejecutar tareas asíncronas en segundo plano (background) mediante el uso de Threads [50]. Si bien, como explicamos anteriormente, hay mensajes asíncronos y síncronos, tuvimos que simular estos últimos por una limitación del lenguaje, ya que a partir de la versión 3.x (Honeycomb) de Android no se pueden hacer request http en el main thread, por lo que se utilizó la clase mencionada anteriormente.

El método doInBackground() de la clase WSTask es el que se encarga de consumir los Web Services a través de la utilización de la librería ksoap2 [46], en la Figura 20 se muestra en detalle el código del método en cuestión para una mayor

comprensión.

```
protected SoapObject doInBackground(Void... params) {

    try {

        SoapManager soapManager = new SoapManager();
        // Modelo el request
        SoapObject request = new SoapObject(soapManager.getNamespace(),
this.getMethodName());

        // Paso parametros al WS
        for(NameValuePair parameter : this.getParameters()) {
            request.addProperty(parameter.getName(), parameter.getValue());
        }

        // Modelo el sobre
        SoapSerializationEnvelope sobre = new SoapSerializationEnvelope(SoapEnvelope.VER11);
        sobre.setOutputSoapObject(request);

        // Modelo el transporte
        HttpTransportSE transporte = new HttpTransportSE(soapManager.getUrl());

        // Llamada
        transporte.call(soapManager.getNamespace() + this.getMethodName(), sobre);

        // Resultado
        SoapObject resultado;
        resultado = (SoapObject) sobre.getResponse();

        return resultado;

    } catch (IOException e) {
        e.printStackTrace();
        Log.e("ERROR_IO", ''+e.toString());
        return null;
    } catch (Exception e) {
        e.printStackTrace();
        Log.e("ERROR", ''+e.getMessage());
        return null;
    }
}
```

Figura 20. Método doInBackground() de la clase WSTask

En la Figura 21 se muestra un fragmento del código del login de los subgrupos donde se instancia un objeto WSTask.

```

...
//obtengo el subgrupo de la vista
String subgrupo = ((TextView) rootView.findViewById(R.id.subgrupo)).getText().toString();

//los agrego para pasarlos como parámetro
nameValuePairs.add(new BasicNameValuePair("subgrupo", subgrupo));

//instancio la clase que ejecuta el web service
WSTask loginTask = new WSTask();
loginTask.setReferer(getActivity());
loginTask.setMethodName(SoapManager.METHOD_LOGIN);
loginTask.setParameters(nameValuePairs);
loginTask.executeTask("completeLoginTask", "errorLoginTask");
...

```

Figura 21. Instanciación del objeto WSTask

Un aspecto importante a tener en cuenta es la necesidad de sincronizar a los distintos subgrupos, ya sea para esperar la llegada de los demás, como para consultar o responder consultas de otros subgrupos.

Cuando un subgrupo llega al punto inicial debe esperar que todos los demás subgrupos lleguen al mismo punto para comenzar el juego. Para simular esta espera, al llegar al punto inicial, en el cliente (subgrupo) se crea un servicio llamado PoolServiceEstados que, cada un intervalo de tiempo determinado (en el prototipo son 5 segundos), envía una solicitud al servidor mediante un método del WebService “preguntando” si todos los demás subgrupos ya llegaron al punto inicial. De ser así, se destruye este servicio y comienza el juego en todos los clientes simultáneamente.

Este comportamiento se repite al finalizar el juego, en donde todos los subgrupos se encuentran en el punto final.

De esta manera se simula la sincronicidad, mediante una barrera, esperando que todos los clientes se encuentren en el mismo estado. Esto se puede apreciar en la Figura 22.

```

private void doServiceWork() {
    // Llamar al WS
    ((JuegoColaborativo) getApplication()).esperarEstadoSubgrupos();
}

```

Figura 22. Método del servicio que invoca al método que chequea la barrera.

El método, en este caso `esperarEstadoSubgrupos()`, simplemente posibilita la comunicación con el web service mediante WSTask como vimos, y llama al método del servidor que chequea la barrera. En caso de éxito, se invoca al método `completeEsperarEstadoSubgrupos()`, que levanta la barrera solo si el resultado que devuelve el servidor es positivo, y toma una decisión al respecto, sino no hace nada y sigue esperando.

En la Figura 23 se ve en detalle, una vez invocado el método explicado, este

toma de un objeto Soap el valor que tiene la información que indica si todos los grupos llegaron a la barrera. El algoritmo chequea este valor, si es negativo no hace nada y retorna para seguir esperando la llegada de los demás, en cambio, si es positivo, es decir, llegaron todos los subgrupos a un estado particular, el método chequea el estado particular al cual llegaron y a partir de allí invoca a uno u otro método. En este caso puede estar esperando para comenzar el juego (invoca al método `comienzoJuego()`) o bien para finalizar el juego (invoca al método `finJuego()`). Es decir como vimos en el modelo en el Capítulo 4, que todos lleguen al punto de encuentro inicial o al punto de encuentro final.

```
public void completeEsperarEstadoSubgrupos(SoapObject result) {
    SoapPrimitive res = (SoapPrimitive) result.getProperty("valorInteger");
    int llegaronSubgrupos = Integer.parseInt(res.toString());
    //chequeo si el valor del resultado es positivo para levantar la barrera
    if (llegaronSubgrupos == 1){
        //dependiendo del estado esperado, el metodo que llama después de ejecutar la tarea
        if (getSubgrupo().getEstado() == getSubgrupo().ESTADO_JUGANDO){
            this.comienzoJuego();
        }else{
            this.finJuego();
        }
    }
}
```

Figura 23. Método `completeEsperarEstadosSubgrupos()`

Por último, tenemos el caso de la comunicación entre dispositivos. En el caso del prototipo, lo resolvimos de la siguiente manera: cuando un subgrupo comienza a jugar, se queda esperando en segundo plano que le lleguen preguntas de los subgrupos conocidos, a través de su comunicación con el servidor. Se instancia un servicio llamado `PoolServiceColaborativo` similar al explicado en la comunicación por barreras. De este modo, cuando hay alguna pregunta sin responder, al chequear cada un intervalo determinado llamando a un método del Web Service, el subgrupo es notificado. Finalmente cuando termina de jugar, el subgrupo detiene el servicio y el servicio es parado.

En la Figura 24 se puede apreciar como instanciar al pool service:

```
//llamo al PoolServiceColaborativo para chequear si un subgrupo realizó una pregunta que debo
responder
this.getCurrentActivity().startService(new Intent(getCurrentActivity(), PoolServiceColaborativo.class));
```

Figura 24. Llamada al pool service colaborativo.

En la Figura 25 se muestra el método invocado cuando se dispara el servicio.

```
private void doServiceWork() {
    // Llamar al WS
    ((JuegoColaborativo) getApplication()).esperarPreguntasSubgrupos();
}
```

Figura 25. Comportamiento del método del pool service colaborativo que invoca al método que espera por preguntas.

En la Figura 26 se puede apreciar en detalle el código del método invocado por el servicio que chequea si existen preguntar por responder, vemos como simplemente el método instancia un objeto WSTask para comunicarse con el servidor invocando al método del web service que chequea lo dicho, la última línea de código, invoca a un método u otro dependiendo si el llamado al Web service falló o no.

```
/*
Método que es llamado por el PoolServiceColaborativo para chequear si hay preguntas por responder
*/
public void esperarPreguntasSubgrupos(){
    ArrayList<NameValuePair> nameValuePairs = new ArrayList<NameValuePair>(1);
    nameValuePairs.add(new BasicNameValuePair("idSubgrupo", Integer.toString(getSubgrupo().getId())));

    WSTask esperarEstadoTask = new WSTask();
    esperarEstadoTask.setReferer(this);

    esperarEstadoTask.setMethodName(SoapManager.METHOD_EXISTE_PREGUNTA_SIN_RESPONDER);
    esperarEstadoTask.setParameters(nameValuePairs);
    esperarEstadoTask.executeTask("completeEsperarPreguntasSubgrupos",
    "errorEsperarPreguntasSubgrupos");
}
```

Figura 26. Detalle del método esperarPreguntasSubgrupos().

De este modo, cuando un subgrupo precisa realizar una consulta al resto, instancia como vimos un objeto WSTask e invoca al método correspondiente.

Es en el servidor donde está la lógica necesaria para responder si un subgrupo espera por respuestas o si hace falta que un subgrupo responda alguna pregunta.

- **Los mapas del prototipo**

La utilización de mapas es un aspecto importante de este juego. En el módulo de administración web, para la configuración de las piezas a recolectar, utilizamos un plugin de jQuery [41] para Google Maps denominado Gmap3 [42].

También nos resultó imprescindible desplegar mapas en los dispositivos a lo largo del transcurso del juego, por tal motivo decidimos utilizar el servicio que proporciona Google Maps para Android [40] que nos proporcionó las librerías

necesarias para darle toda la funcionalidad al prototipo.

De este modo logramos disponer de librerías que nos facilitaron la utilización de la georreferenciación, la creación y customización de markers [43], el conocimiento de la ubicación de los dispositivos mediante el GPS, y todo lo relacionado al posicionamiento.

- **Los sensores de proximidad**

Existe en el juego una interacción continua con puntos de encuentro y piezas a recolectar. En el prototipo estas se despliegan a través de markers [43] customizados. Además, la posición de los subgrupos se detecta a través de los datos GPS de cada dispositivo, y en cada POI decidimos utilizar sensores de proximidad para indicar automáticamente la llegada de un subgrupo a un punto de encuentro o a una pieza a recolectar.

En la Figura 27 se puede apreciar fragmentos de código donde se especifica lo antes mencionado. En la figura vemos como a un POI determinado, en este caso el punto inicial, que contiene la información posicionamiento para ubicarlo en el mapa, se le agrega una alerta de proximidad (`addProximityAlert()`) y finalmente se lo agrega como marker al mapa del juego, especificando las coordenadas, el título y la imagen customizada (una bandera azul como veremos en la simulación).

```
Poi puntoInicial = subgrupo.getPoiInicial();
addProximityAlert(puntoInicial, PROX_ALERT_PUNTO_INICIAL, 0);
this.getGoogleMap().addMarker(new MarkerOptions().position(new
    LatLng(puntoInicial.getCoordenadas().getLatitud(),
    puntoInicial.getCoordenadas().getLongitud())).title("Punto
    inicial").icon(BitmapDescriptorFactory.fromResource(R.drawable.start_flag)));
```

Figura 27. Los proximity alerts.

En la Figura 28 se detalla el método que agrega una alerta de proximidad a una ubicación determinada de la clase POI. a través de la utilización de librerías específicas de android, como *Intent*, *IntentFilter*, *Location*, *LocationManager* y una clase que creamos que extiende de *BroadcastReceiver*, se parametriza el sensor según las características del prototipo del juego, por ejemplo el radio de proximidad, en nuestro caso pusimos un radio grande al estar falseando las ubicaciones, se puede achicar hasta un metro, o también la expiración de la alerta, para el prototipo no es necesario que expire, simplemente la detendremos cuando sea necesario.

```
private void addProximityAlert(Poi poi, String intentName, int extraParameter) {
    Intent intent = new Intent(intentName);
    intent.putExtra("id", extraParameter);

    PendingIntent proximityIntent = PendingIntent.getBroadcast(this, 0, intent,
    PendingIntent.FLAG_UPDATE_CURRENT);
```



```

locationManager.addProximityAlert(
    poi.getCoordenadas().getLatitud(), // the latitude of the central point of the alert region
    poi.getCoordenadas().getLongitud(), // the longitude of the central point of the alert region
    POINT_RADIUS, // the radius of the central point of the alert region, in meters
    PROX_ALERT_EXPIRATION, // time for this proximity alert, in milliseconds, or -1 to indicate no expiration
    proximityIntent // will be used to generate an Intent to fire when entry to or exit from the alert region is
    detected
);

IntentFilter filter = new IntentFilter(intentName);
ProximityIntentReceiver proximityIntentReceiver = new ProximityIntentReceiver();
proximityIntentReceiver.setApplication(((JuegoColaborativo) getApplication()));
registerReceiver(proximityIntentReceiver, filter);
}

```

Figura 28. Detalle del método addProximityAlert().

En las últimas líneas de la Figura 28 se observa cómo se instancia un objeto *ProximityIntentReceiver*, que será el encargado de tomar una acción cuando el dispositivo esté dentro del radio del sensor.

El método onReceive() de la clase *ProximityIntentReceiver* que extiende de *BroadcastReceiver* es el que alertará cuando los datos proporcionados por el GPS de la aplicación entren al radio de uno de los sensores. Cada *intent* creado anteriormente posee un nombre o acción, entonces a partir de esa acción, el algoritmo decide que método invocar, pueden ser tres, llegada al punto de encuentro inicial, final o a alguna de las piezas a recolectar.

Finalmente cuando es invocado alguno de los métodos disparados por la alerta de proximidad, debemos detenerla, ya que no será utilizada nuevamente y puede crear conflictos si el dispositivo vuelve a pasar por el mismo lugar, y tomar las acciones correspondientes según el POI donde se encuentra el subgrupo.

A través nuevamente de la acción del intent se invoca en algún momento al método removeProximityAlert() que simplemente detiene la alerta, este está especificado en la Figura 29.

```

public void removeProximityAlert(String intent) {
    String context = Context.LOCATION_SERVICE;
    LocationManager locationManager = (LocationManager) getSystemService(context);

    Intent anIntent = new Intent(intent);
    PendingIntent operation = PendingIntent.getBroadcast(getApplicationContext(), 0, anIntent, 0);
    locationManager.removeProximityAlert(operation);
}

```

Figura 29. Método removeProximityAlert()

De este modo, resolvemos de forma automática y transparente para los participantes, los eventos disparados a partir de su llegada a ubicaciones específicas, como por ejemplo a los puntos de encuentro o a las piezas a recolectar. Existen otras alternativas tales como a partir de la llegada a una ubicación, localizar

un código QR [57], leerlo con el dispositivo y que esa lectura dispare algún evento. Aunque dejaría de ser automático y es necesaria la instalación de aplicaciones de lectura de códigos QR en cada uno de los dispositivos clientes.

5.2 Dificultades técnicas en el desarrollo

A lo largo del desarrollo de la tesina hemos enfrentado algunas dificultades técnicas, a continuación enumeraremos las que creemos más relevantes:

- **Complejidad de los Web Services:** Como explicamos anteriormente, no fue simple establecer la comunicación entre cliente y servidor, más aún cuando utilizamos diferentes tecnologías para implementar uno y otro. Lo resolvimos utilizando el protocolo SOAP para estandarizar la comunicación a través de intercambio de datos XML [47], del lado del servidor utilizamos un plugin de Symfony [45], que nos automatizó la generación del archivo WDSL [51] necesario para la implementación de Web Services. Por el lado del cliente lo resolvimos utilizando la librería ksoap2 de android [46] que utiliza SOAP para realizar las llamadas a algunas de las funciones listadas en el WDSL. Nos llevó un tiempo poner en funcionamiento estas tecnologías en conjunto y familiarizarnos con ellas, pero una vez resuelto los primeros intercambios de mensajes entre cliente y servidor, se nos facilitó notablemente la comunicación entre ambos.
- **Idas y vueltas de las keys de desarrollo de gmaps:** El desarrollo de los mapas en el cliente Android se realizó utilizando la API de google maps para Android [40], uno de los requerimientos para utilizar esta API, es obtener un certificado Android y la clave de la API de Google maps. Si bien todo esto no tiene costo económico, no es fácil obtener esta clave de Google Maps para un entorno de desarrollo y que esta funcione rápidamente. Usualmente lleva un tiempo hacer que todo funcione, no es fácil saber qué está fallando cuando un mapa no se despliega, por ejemplo. Una restricción de esta clave de desarrollo, es el límite de 25000 request diarios, aunque es muy difícil llegar a ese número.
- **Posicionamiento fake (falso) en pruebas:** Como vimos, el juego tiene un fuerte componente de posicionamiento, es necesario conocer la posición actual de cada subgrupo y a medida que cambia de posición, alertar sobre la proximidad de algún poi, por ejemplo. Ahora bien, una vez resuelta la dificultad técnica que esto implica, a la hora de probar el funcionamiento juego, es imposible que lo hagamos posicionándonos físicamente en los lugares establecidos. Por este

motivo necesitamos de una herramienta para dispositivos android que simule una ubicación GPS, de modo que para nuestro juego (y para cualquier aplicación que utilice el GPS del teléfono), el subgrupo que utiliza ese dispositivo, sea transparente y realmente crea que uno está situado allí. Probamos varias de estas aplicaciones disponibles y finalmente elegimos la que mejor funcionaba para nuestras pruebas y dinámica del juego (Fake gps - fake location [52])

- Limitaciones en los emuladores de dispositivos móviles: Si bien existen varias opciones para el desarrollo y posterior ejecución de una aplicación Android en un emulador, nos fue dificultoso utilizarlos por las limitaciones propias de los mismos. Habiendo probado el emulador por defecto de la librería SDK de Android [39] e inclusive un emulador gratuito (Genymotion) [53] que parecía mejorar la performance, la velocidad de ejecución de la aplicación era menos que aceptable, por lo que todas las pruebas las debimos hacer en dispositivos reales. Otra limitación de los emuladores, aparte de la velocidad de ejecución, fue la imposibilidad de utilizar la API de Google Maps.

5.3 Administración Web

Comencemos a describir el módulo de administración web que hemos desarrollado, el primer paso para entrar al sistema es el login, para dar seguridad básica a la aplicación. Este se puede visualizar en la Figura 30.

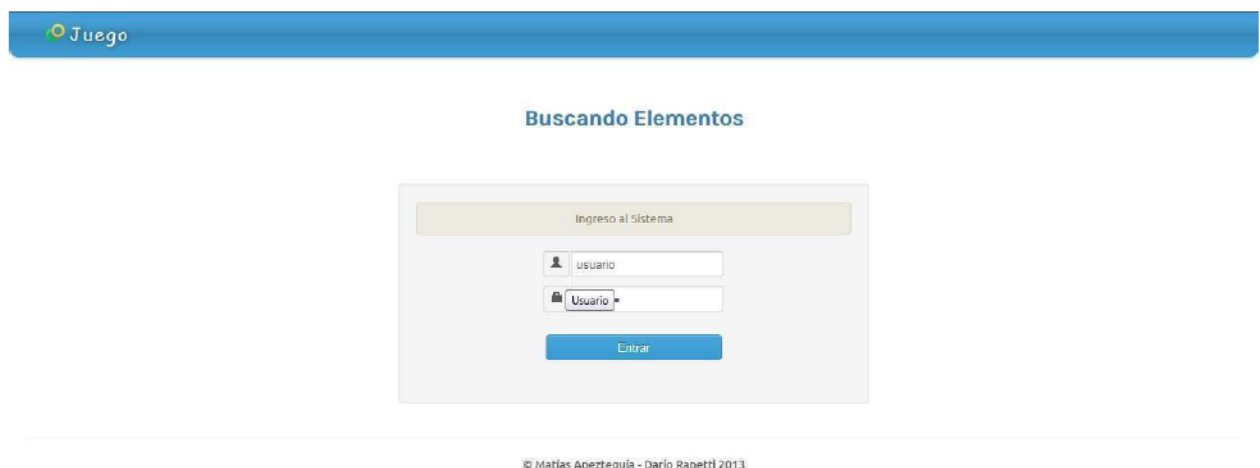


Figura 30. Login de la aplicación web

Una vez logueado, la primera pantalla que se muestra es la de los grupos, con la posibilidad de ver los datos principales, acceder al detalle, editar, eliminar o crear nuevos grupos. Esto se puede visualizar en la Figura 31.



© Matías Apezteguía - Darío Rapetti 2013

Figura 31. Pantalla de los grupos.

Como se ve en la Figura 31 a la izquierda está el menú con los atajos a los principales ítems de configuración. En la Figura 32 se muestra este menú ampliado para que se visualice mejor.

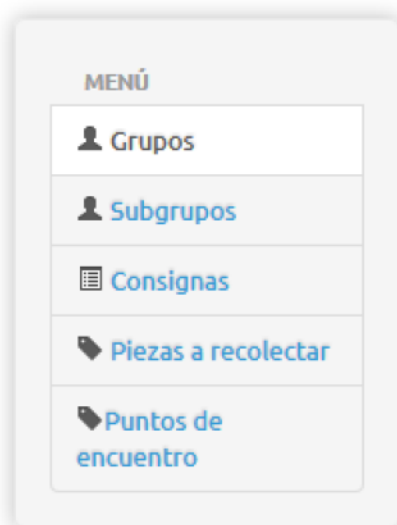


Figura 32. Menú del juego.

Continuemos con el grupo, veamos ahora, la creación de nuevos grupos y subgrupos. Lo decidimos realizar en un solo paso, para simplificar la configuración al usuario, primero completa los campos del grupo, tales como su nombre y su consigna asociada, si es que la tiene, sino la completa más adelante. Luego tiene la opción de configurar los subgrupos. En la Figura 33 se muestra la pantalla de creación.

Crear Grupo

Nombre:

Consigna: Animales mamíferos

Subgrupos

Agregar subgrupo: ☒

Nombre	Acciones
No hay datos para mostrar	

Figura 33. Alta de grupos y subgrupos.

En la figura anterior se puede apreciar cómo se completan los datos del grupo, y debajo los de los subgrupos, veamos en profundidad cómo se van agregando los subgrupos, primero se espera que se complete su nombre (ver Figura 34).

Subgrupos

Agregar subgrupo: ☒

Figura 34. Alta de subgrupos, dentro del alta de grupos

Una vez tildado el nombre elegido, se agrega el subgrupo y aparecen las opciones para ir agregando uno a uno a los participantes involucrados en el juego (ver Figura 35).

Subgrupos

Agregar subgrupo: ☒

Nombre	Acciones
Subgrupo 1	<input type="button" value="✎"/> <input type="button" value="🗑"/>

Participantes

Nombre	Edad	Acciones
No hay datos para mostrar		

Figura 35. Subgrupos y participantes

Como vemos en la figura anterior se pueden seguir creando nuevos subgrupos como lo explicamos, o bien ir creando uno a uno a los participantes, simplemente pulsando el botón amarillo “Nuevo participante”. En la Figura 36 se puede apreciar la pantalla que sale al seleccionar la opción mencionada.

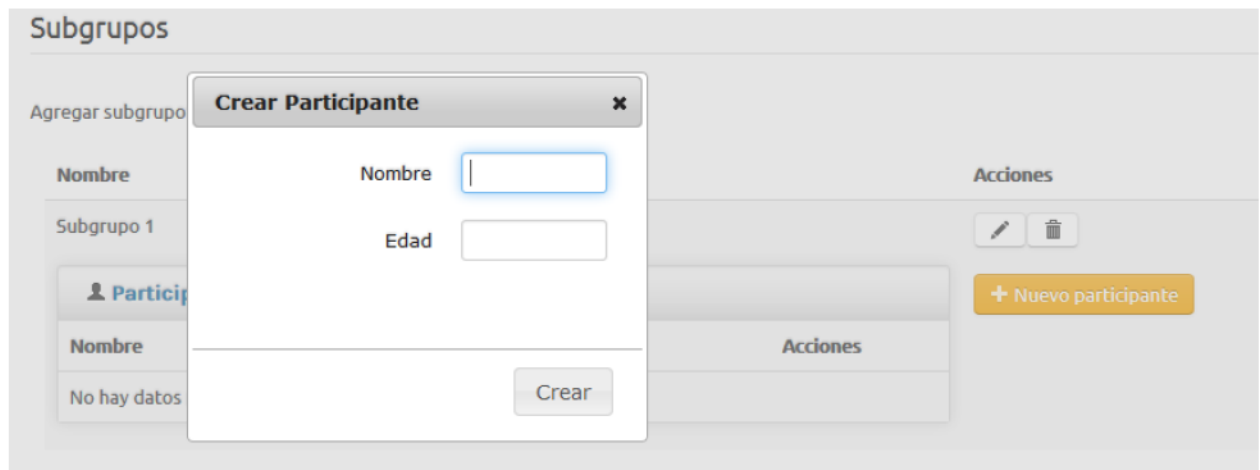
The image shows a web application interface with a modal window titled "Crear Participante" (Create Participant) in the center. The modal has a close button (X) in the top right corner. Inside the modal, there are two input fields: "Nombre" (Name) and "Edad" (Age). The "Nombre" field is currently active, indicated by a blue border. Below the input fields is a "Crear" (Create) button. In the background, a sidebar on the left is partially visible with the heading "Subgrupos" and a list of subgroups. The main content area on the right has a table with columns "Nombre" and "Acciones". The "Acciones" column contains icons for editing and deleting. A yellow button labeled "+ Nuevo participante" (New participant) is located at the bottom right of the main content area.

Figura 36. Creación de participantes

Así de esta forma se pueden ir creando uno a uno nuevos participantes y subgrupos. De esta forma logramos unificar la creación de grupos, subgrupos y participantes intentando que la carga sea lo más dinámica posible. En la Figura 36 se puede apreciar un grupo creado, con sus subgrupos y los participantes de los mismos.

Se creó correctamente el participante

[✎ Crear Grupo](#)

Nombre

Consigna

Subgrupos

Agregar subgrupo:

Nombre

Subgrupo 1

[👤 Participantes](#)

Nombre	Edad	Acciones
Fede	9	<input type="button" value="✎"/> <input type="button" value="🗑"/>
Florencia	11	<input type="button" value="✎"/> <input type="button" value="🗑"/>
Juan	10	<input type="button" value="✎"/> <input type="button" value="🗑"/>

Subgrupo 2

[👤 Participantes](#)

Nombre	Edad	Acciones
Carla	11	<input type="button" value="✎"/> <input type="button" value="🗑"/>
Joaquin	10	<input type="button" value="✎"/> <input type="button" value="🗑"/>
Maitena	11	<input type="button" value="✎"/> <input type="button" value="🗑"/>

Acciones

Figura 37. Creando grupos, subgrupos y participantes.

Como se observa en la Figura 37 se está creando el “Grupo Azul”, con la consigna “Animales Mamíferos” (que ya existe previamente y será explicada su creación más adelante) y además se crearon dos subgrupos, con tres participantes cada uno. Cabe destacar que en todo momento se pueden corregir malas cargas fácilmente durante o después de haber creado los grupos, para esto se pueden utilizar los botones mostrados en la Figura 38.



Figura 38. Botones para editar o eliminar subgrupos y/o participantes.

Al volver a la pantalla principal de los grupos, el sistema avisa mediante un mensaje que no hubo inconvenientes en la modificación realizada y muestra el nuevo grupo en la grilla (ver Figura 39).



Figura 39. Pantalla principal de los grupos con el nuevo grupo creado.

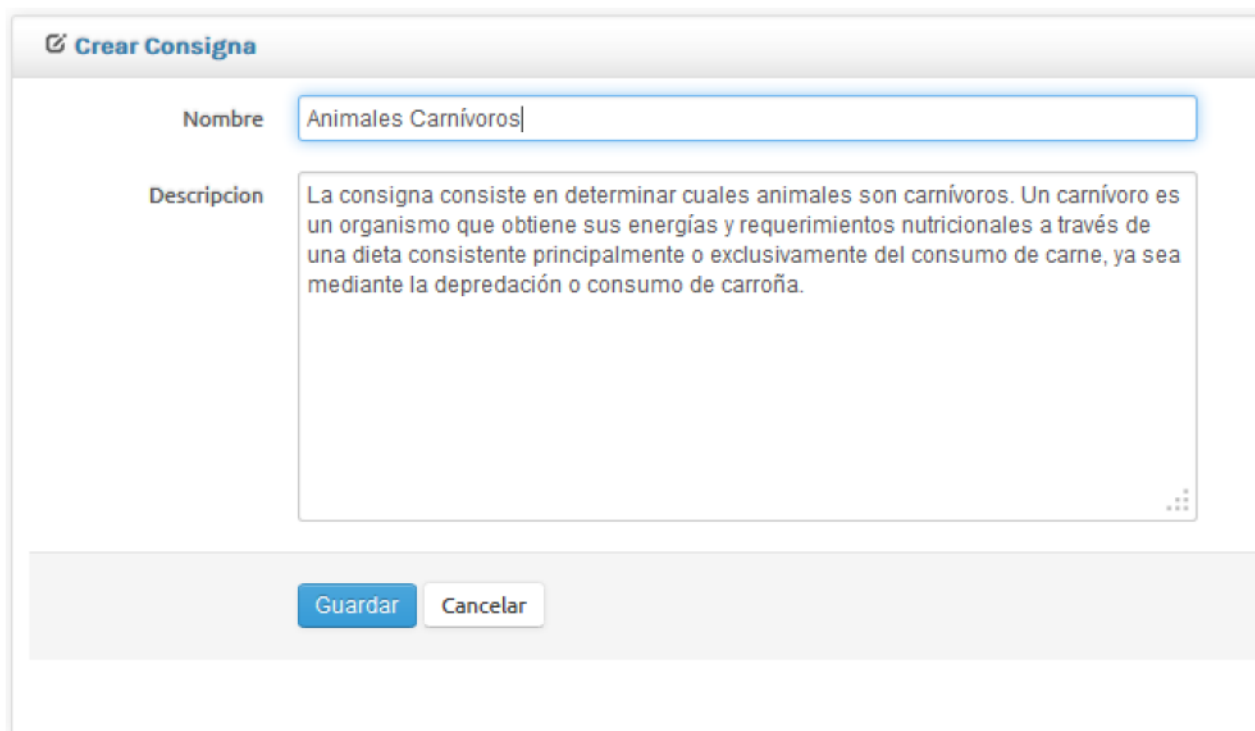
Ahora bien, continuemos con la creación de consignas, posee como los grupos una pantalla inicial con una grilla y los datos principales (como se muestra en la Figura 40). Y luego una interfaz simple para crear nuevas consignas.



© Matías Apezteguía - Darío Rapetti 2013

Figura 40. Pantalla principal de las consignas.

En la Figura 41 se muestra la pantalla de creación de consignas, muy simple, son solo dos campos a completar, el nombre y la descripción, tener en cuenta que luego serán de mucha utilidad para los participantes a la hora de decidir si una pieza cumple o no con la consigna.



Crear Consigna

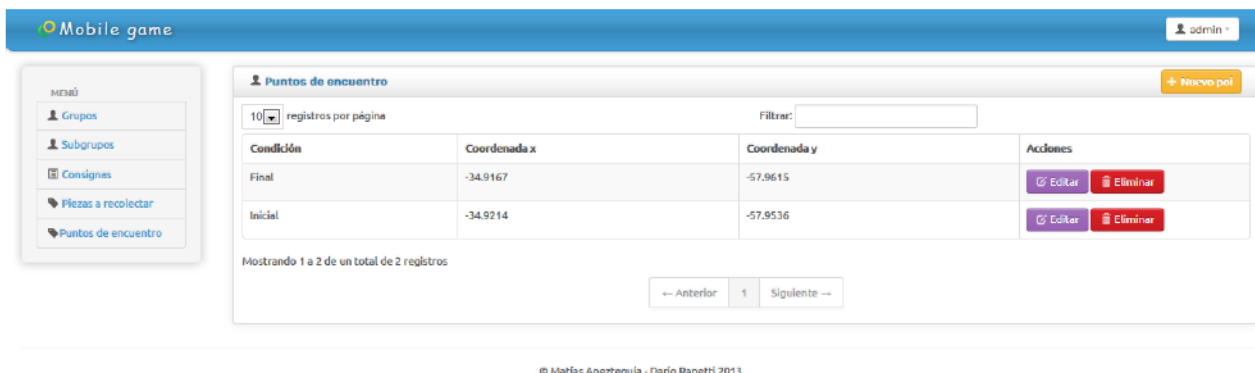
Nombre: Animales Carnívoros

Descripción: La consigna consiste en determinar cuales animales son carnívoros. Un carnívoro es un organismo que obtiene sus energías y requerimientos nutricionales a través de una dieta consistente principalmente o exclusivamente del consumo de carne, ya sea mediante la depredación o consumo de carroña.

Guardar Cancelar

Figura 41. Creación de consignas.

Continuemos ahora con la creación del punto de encuentro inicial y final del juego (ver Figura 42). En el modelo inicial los puntos de encuentro poseen una descripción, para la implementación del prototipo se obvió este atributo, ya que no hace al objetivo de esta implementación.



Mobile game admin

Puntos de encuentro + Nuevo poi

10 registros por página Filtrar:

Condición	Coordenada x	Coordenada y	Acciones
Final	-34.9167	-57.9615	Editar Eliminar
Inicial	-34.9214	-57.9536	Editar Eliminar

Mostrando 1 a 2 de un total de 2 registros

Anterior 1 Sigiente

© Matías Apezteguía - Darío Rapetti 2013

Figura 42. Pantalla principal de los puntos de encuentro.

En la Figura 42 se pudo apreciar, como en todas las pantallas principales, una grilla con los datos principales, en este caso de los puntos de encuentro con sus respectivas acciones, la edición, eliminación o creación de nuevos puntos de encuentro.

Es necesaria una aclaración, en el modelo inicial, cada POI, en este caso los asociados a los puntos de encuentro, están relacionados a una *Posicion*. Nosotros en este prototipo hemos decidido llevarla a cabo mediante coordenadas. En la

Figura 43 se muestra como es la creación de lo antes mencionado.

Figura 43. Creación de Puntos de encuentro.

Explicaremos en detalle la creación de puntos de encuentro, primero se completa especificando y se está creando el punto de encuentro inicial o final.

Luego es necesario posicionar el punto de encuentro en el mapa. Tenemos dos opciones, si conocemos las coordenadas, escribirlas, o sino posicionarse con el mapa en el lugar deseado y apretar el botón derecho del mouse, se le desplegará una ventana donde se le dará la opción de generar allí la posición (Figura 44).



Figura 44. Elegir la posición del punto de encuentro desde el mapa

Una vez elegida la posición, además de mostrarse en el mapa, se actualizarán los campos de las posiciones, como se visualiza en la Figura 45.

Ubicación del punto de encuentro

Condición

Latitud

Longitud

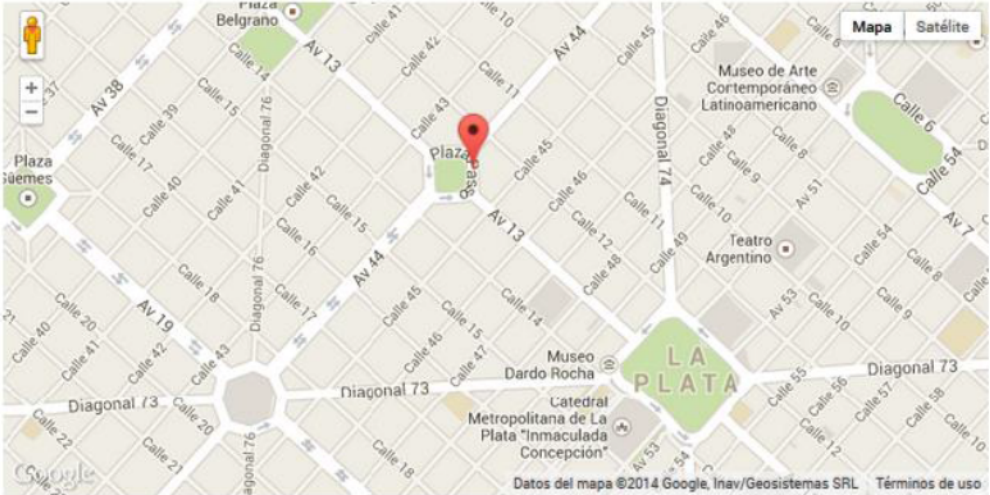


Figura 45. Configurando la posición del punto de encuentro.

Hasta ahora hemos creado los grupos, subgrupos, participantes y consignas. Continuemos con las piezas a recolectar. La forma de ubicarlas en el mapa es similar a la de los puntos de encuentro. Ya que en realidad, como vimos en el modelo, lo que se ubica en el mapa son los POIS a través de las coordenadas como forma de posicionamiento elegida. En la Figura 46 se muestra la lista de piezas a recolectar creadas.

Piezas a recolectar Ver Todas + Nueva pieza a recolectar

10 registros por página Filtrar:

Nombre	Descripción	Ubicación	Consignas	Acciones
Murciélago	Sus extremidades superiores se desarrollaron como alas. Poseen aproximadamente mil cien especies.	-34.9203 , -57.96	- Animales mamíferos - Animales Carnívoros	Detalle Editar Eliminar
Perro	Constituye una subespecie del lobo (Canis lupus). Su tamaño o talla, su forma y pelaje es muy diverso según la raza. Posee un oído y olfato muy desarrollados, siendo este último su principal órgano sensorial.	-34.9206 , -57.9622	- Animales mamíferos - Animales Carnívoros	Detalle Editar Eliminar
Pinguino	Los pingüinos viven en el mar abierto del Hemisferio Sur; son los únicos animales vivos no voladores adaptados al buceo propulsado por las alas	-34.9113 , -57.9427	- Animales Carnívoros	Detalle Editar Eliminar
Tortuga	Forman un orden de reptiles caracterizados por tener un tronco ancho y corto, y un caparazón que protege los órganos internos de su cuerpo. De su caparazón salen, por delante, la cabeza y las patas delanteras, y por detrás las patas traseras y la cola.	-34.9203 , -57.9498		Detalle Editar Eliminar

Mostrando 1 a 4 de un total de 4 registros

[← Anterior](#) [1](#) [Sigüente →](#)

Figura 46. La pantalla inicial de piezas a recolectar

En la Figura 46, podemos observar como aparece la información en una grilla de cada pieza, su nombre descripción, ubicación y sus consignas asociadas si es que tiene alguna. De este modo se puede ver claramente a simple vista o utilizando los filtros, las piezas a las que pertenece cada consigna.

Veamos ahora cómo crear una nueva pieza a recolectar (Figura 47).

Crear Pieza a recolectar

Nombre:


Descripción:

Consignas:

Ubicación de la pieza

Latitud:

Longitud:



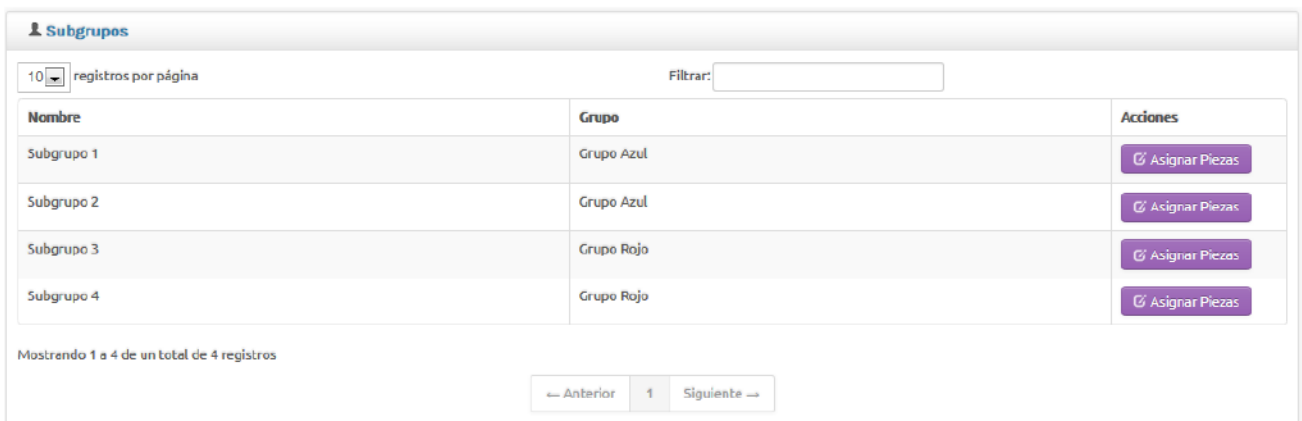
[Guardar](#) [Cancelar](#)

Figura 47. Creación de piezas a recolectar

Como dijimos, la forma de especificar la posición es similar a los puntos de encuentro, también se debe especificar un nombre de la pieza y una descripción, que será muy útil para los participantes a la hora de tomar una decisión en base a la consigna de cada grupo. Por último en la creación de piezas se especifica si cumple con una consigna, con varias o con ninguna.

Solo resta, de acuerdo a la dinámica del juego escogido asignar cuáles piezas deberá recorrer cada subgrupo de un grupo determinado. Es decir si en el juego existen ocho piezas a recolectar, y por ejemplo un grupo posee dos subgrupos. Debería asignarle cuatro consignas a cada subgrupo, de manera tal que cada subgrupo recorra y tome una decisión sobre sus piezas a recolectar y entre los subgrupos del mismo grupo recorran la totalidad de las piezas del universo del juego. De acuerdo a la consigna del grupo, el juego podrá determinar si las decisiones tomadas fueron las correctas.

En la Figura 48 se muestra la pantalla principal de los subgrupos, en esta además de la información de cada uno se observa un botón violeta para asignar las piezas.



Nombre	Grupo	Acciones
Subgrupo 1	Grupo Azul	Asignar Piezas
Subgrupo 2	Grupo Azul	Asignar Piezas
Subgrupo 3	Grupo Rojo	Asignar Piezas
Subgrupo 4	Grupo Rojo	Asignar Piezas

Mostrando 1 a 4 de un total de 4 registros

← Anterior 1 Siguiente →

Figura 48. Pantalla principal de los subgrupos

La Figura 49 surge luego de presionar el botón para asignar piezas a un subgrupo, se visualiza cómo se pueden ir asignando las piezas a un subgrupo, recordemos que el Subgrupo 1, perteneciente al Grupo Azul.

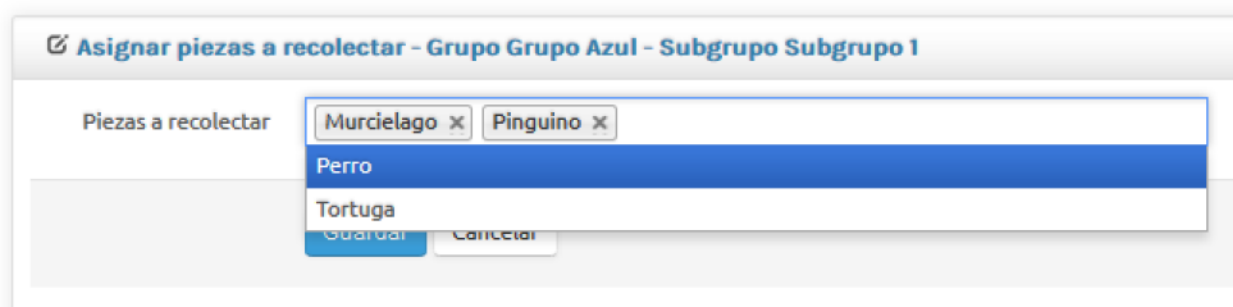


Figura 49. Asignación de piezas a un subgrupo

Por último, para tener una visión general de las piezas del juego, al pulsar el botón “Ver Todas” (Figura 50) en la pantalla inicial de piezas.

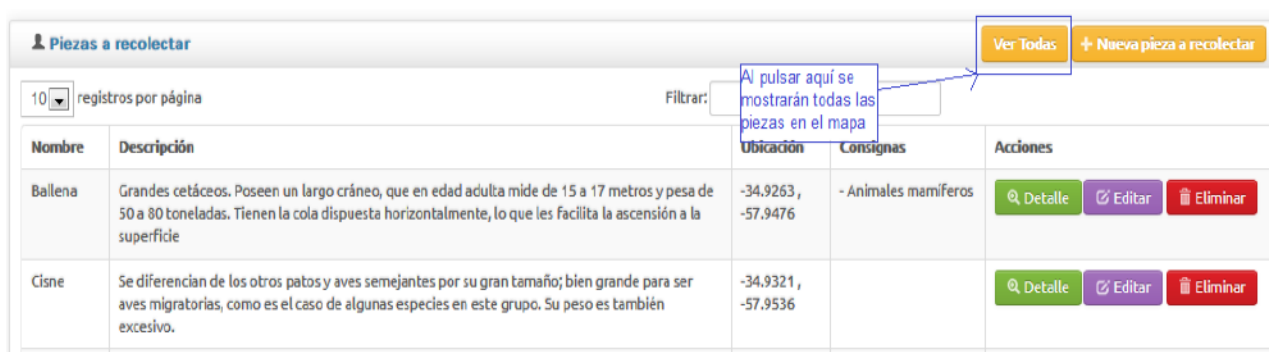


Figura 50. Botón para mostrar todas las piezas a recolectar

Se muestra en pantalla un mapa, con los puntos de encuentro y todas las piezas existentes (Figura 51).

6. Simulación de un Juego Educativo Móvil Colaborativo

Ahora simularemos un ejemplo del juego, a modo de aclaración, el juego ya está desarrollado y funcionando, lo que haremos es configurarlo como se mencionó anteriormente y jugarlo. Explicaremos su funcionamiento a partir de capturas de pantallas del juego en funcionamiento.

Para la simulación del ejemplo, hemos tomado como base parte de las configuraciones hechas anteriormente y hemos añadido las faltantes. Algunas consideraciones:

- Se jugará entre dos grupos, el grupo Azul y el grupo Rojo
- Cada grupo tendrá dos subgrupos asociados, subgrupo 1 y 2 para el grupo Azul y subgrupo 3 y 4 para el grupo Rojo.
- Los dos grupos tendrán asignada la misma consigna, “Animales mamíferos”
- Cada grupo deberá recorrer ocho piezas a recolectar; cuatro que cumplen la consigna (perro, murciélago, elefante y ballena) y cuatro que no (pingüino, tortuga, cisne y surubí).
- Cada subgrupo tendrá entonces asignadas cuatro piezas a recolectar, en este caso dos que cumplen la consigna y dos que no cumplen.
- Simularemos jugar con un subgrupo, el subgrupo 1 del grupo Azul. De este modo interactuaremos con el subgrupo 2. Del grupo Rojo solo tendremos noticias al finalizar el juego, a la hora de ver los resultados. Aunque estarán presentes al momento de esperar que todos lleguen al punto inicial y final.
- Cada pieza a recolectar estará situada en alguna plaza de la ciudad de la plata. Solo a efectos de simplificar la simulación. El juego está pensado como para ofrecer una especie de realidad aumentada, que la consigna y las piezas a recolectar tengan una relación directa con el lugar. Por ejemplo para la simulación de la consigna de animales mamíferos, las piezas podrían estar en lugares donde haya alguna referencia al animal a estudiar, como museos, zoológicos, veterinarias, etc.

Para intentar hacer esta simulación lo más dinámica posible, haremos todo con capturas de pantalla, paso a paso, explicando brevemente qué es lo que está aconteciendo.

Comencemos a jugar. Como primer paso, cada subgrupo debe relacionar su dispositivo con el subgrupo, y entonces que cada aplicación nativa conozca el subgrupo con el que juega. A partir de un simple login con el nombre del subgrupo realizamos esta asociación (ver Figura 52).

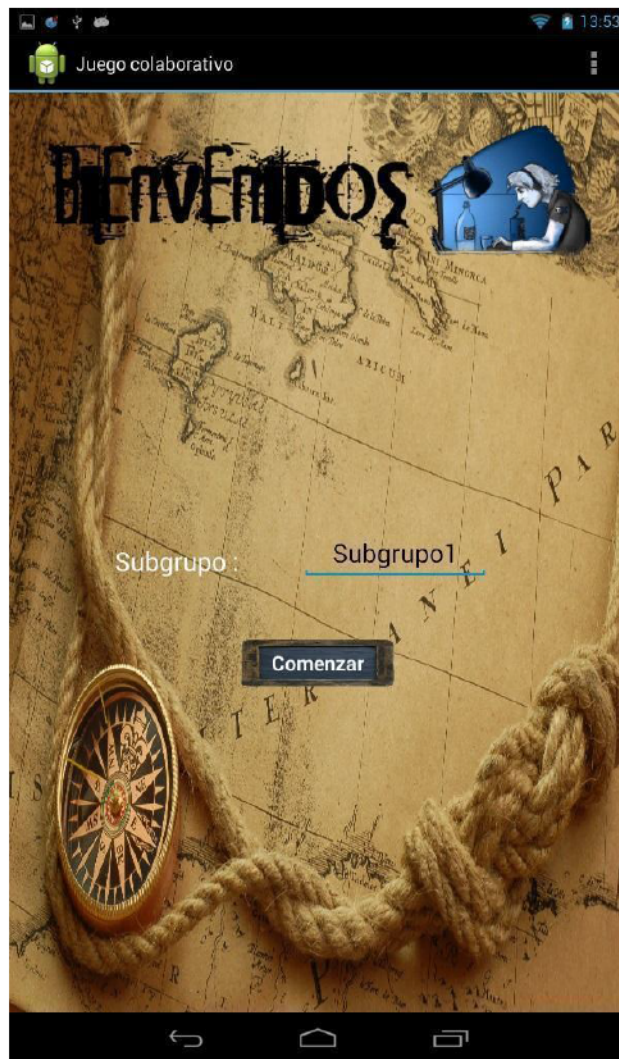
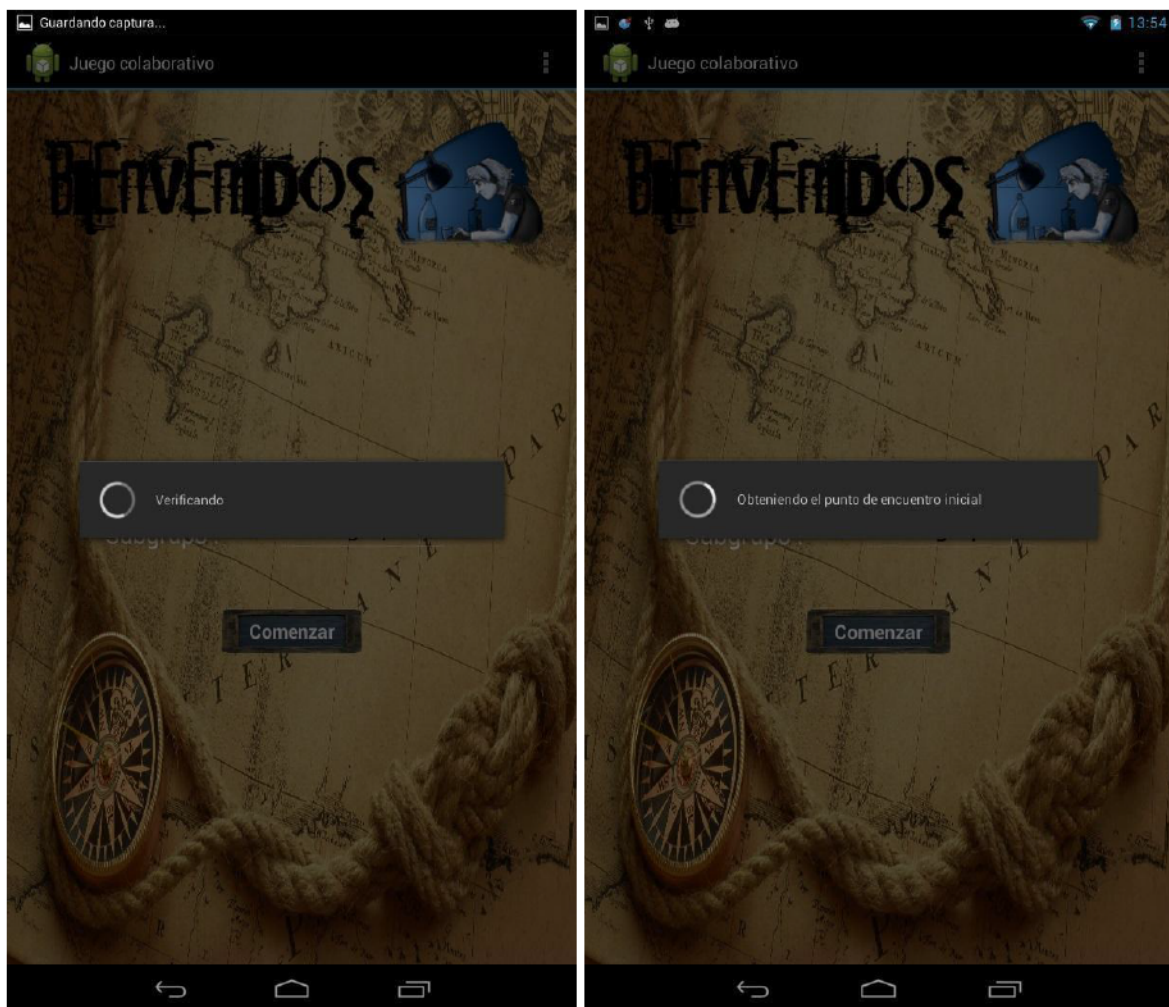


Figura 52. Login del subgrupo

Como vemos, en este prototipo decidimos que no era necesario por ahora utilizar una contraseña para loguearse, de todos modos el modelo está preparado para que se utilice si es necesario.

Al intentar loguearse, la aplicación hará los chequeos correspondientes (Figura 53 y 54) y una vez verificado, asociará al dispositivo con el subgrupo e intentará obtener el punto de encuentro inicial. Al participante se le irán mostrando mensajes de lo que está aconteciendo. Si el login falla, simplemente devuelve un mensaje de error y vuelve a la pantalla inicial.



Figuras 53 y 54. Verificando login y obteniendo punto de encuentro inicial.

El paso siguiente es la aparición del punto de encuentro inicial en un mapa, como se muestra en la Figura 55, este se encuentra en el centro de la plaza Moreno de La Plata. Lo que deben hacer los participantes ahora es acercarse al punto de encuentro.

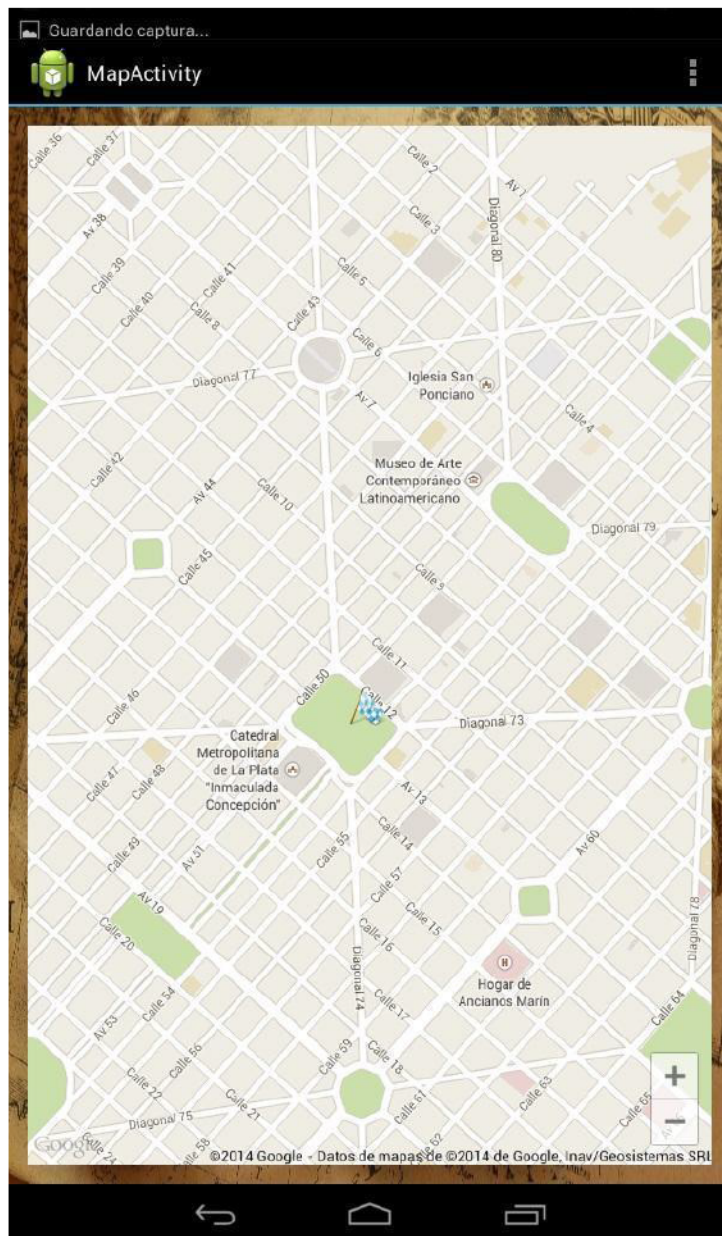


Figura 55. Punto de encuentro inicial

Cuando el subgrupo llega al punto de encuentro, automáticamente, gracias al sensor de proximidad, la aplicación es notificada y muestra a los participantes el mensaje como se puede ver en la Figura 56, ahora simplemente deben esperar que todos los demás subgrupos lleguen.

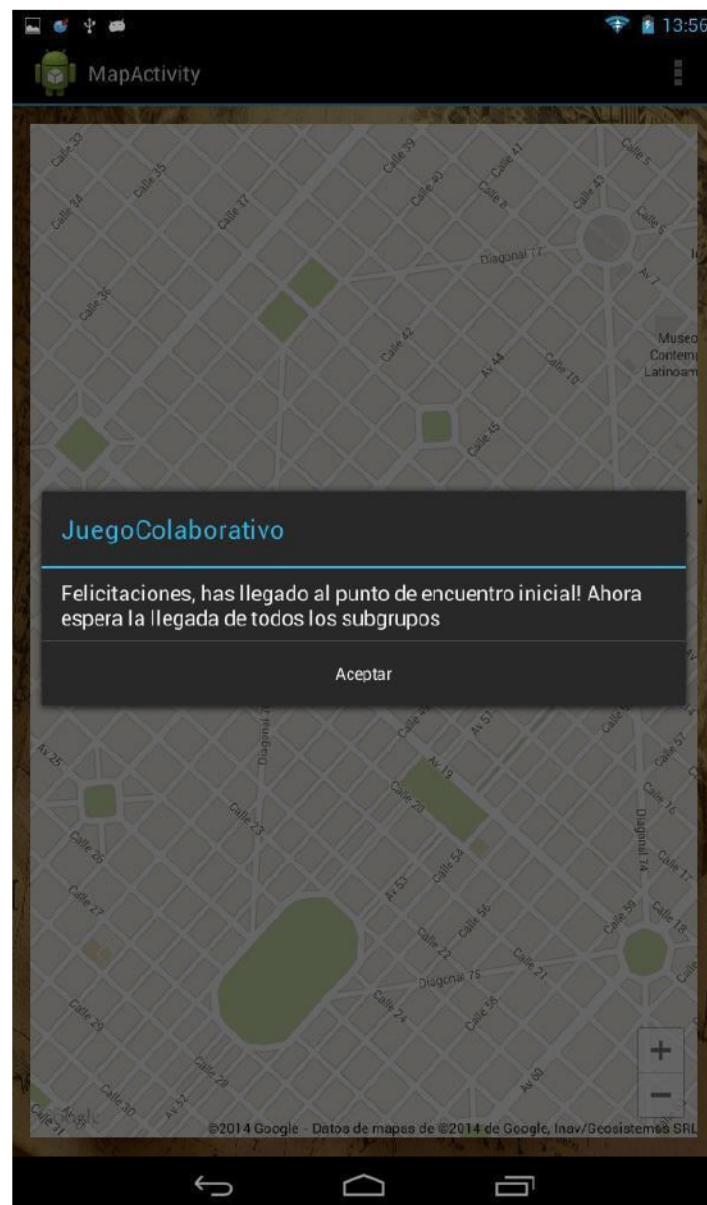


Figura 56. Llegando al punto de encuentro inicial

Al llegar el último subgrupo, la aplicación les avisa que llegaron todos. Y a partir de ahora comienza la etapa de recolección de piezas y consultas a los compañeros (ver Figura 57).

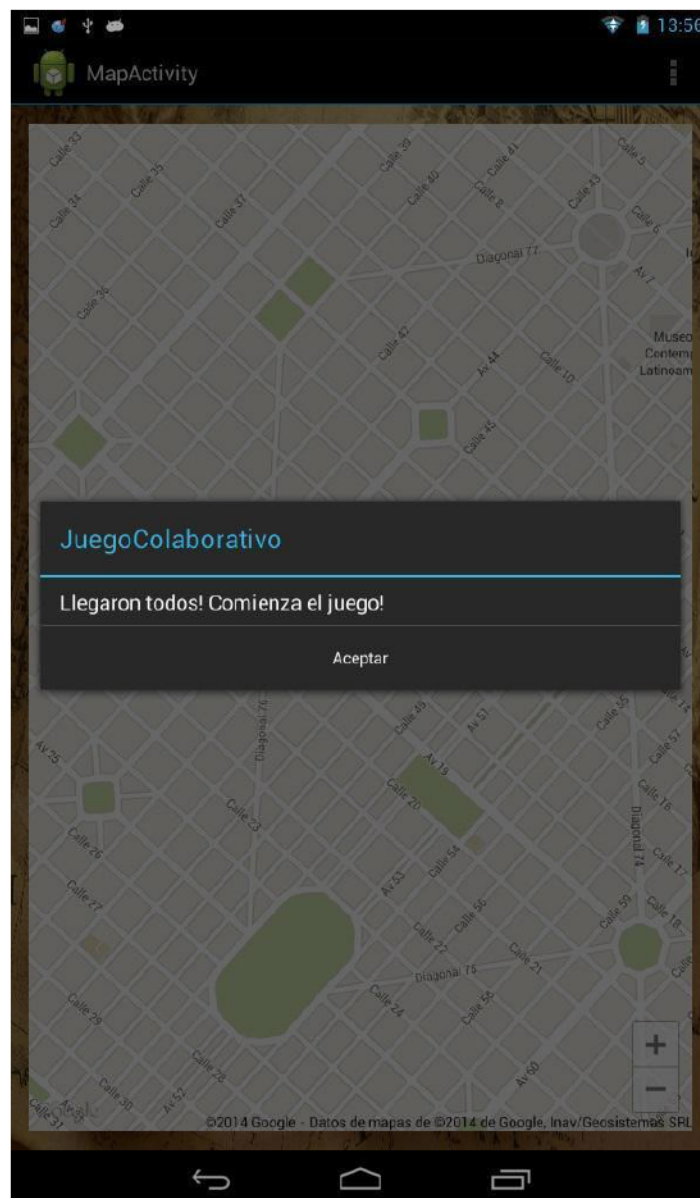


Figura 57. Llegada de subgrupos al punto de encuentro inicial.

En la Figura 58 se puede ver como aparecen en el mapa las cuatro piezas que se deben recolectar y tomar una decisión. Recordemos que cada subgrupo tiene cuatro piezas cada uno para recolectar. Lo primero que deben hacer los grupos es ponerse de acuerdo para ir a una de estas piezas, por ejemplo, elegir la que está en proximidades de la plaza Paso en las avenidas 13 y 44 y dirigirse hacia allá.

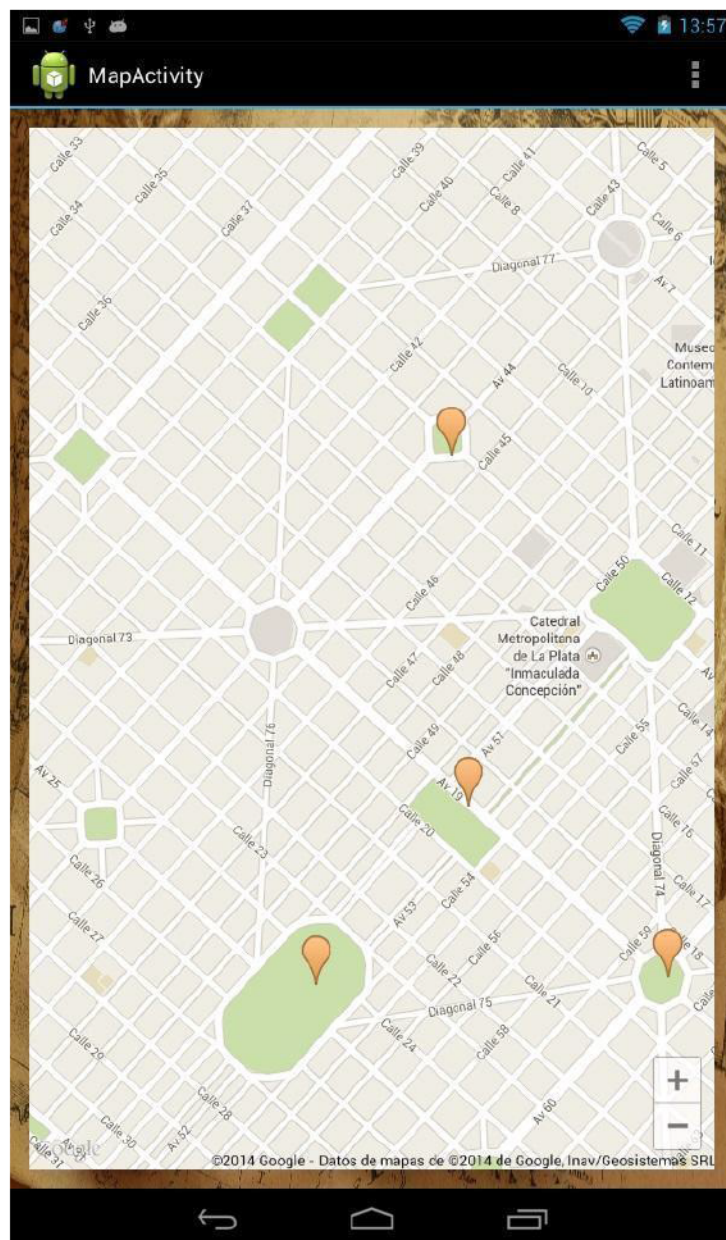


Figura 58. Aparición de piezas a recolectar

Una vez que el grupo al lugar indicado en el mapa, se nos desplegará una pantalla con los datos de la pieza en cuestión. Como vemos en la Figura 59, el encabezado es un mensaje de bienvenida a la pieza, luego una sección donde se nos muestra el nombre y descripción de la consigna, que es general a todo el grupo. Lo que sigue es el nombre y descripción de la pieza a recolectar, en este caso “perro”. Aparece un botón donde decidir si la pieza cumple o no la consigna, y un espacio donde describir nuestra justificación.

Por último vemos dos botones, uno para enviar la respuesta si el subgrupo está seguro, y otro para realizar una consulta a nuestros compañeros. En este caso, como vemos en la figura, los participantes están seguros de su respuesta y deciden enviarla sin consultar.

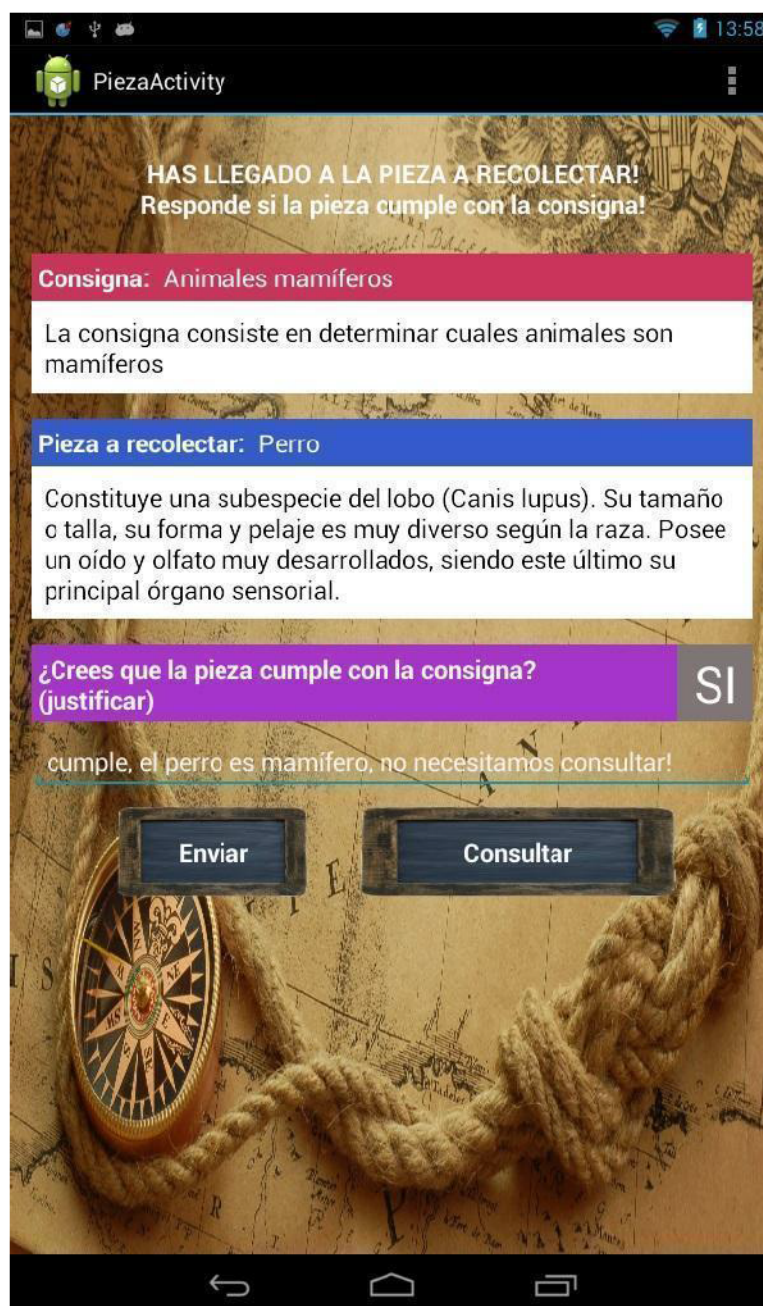


Figura 59. Primera pieza a recolectar

La Figura 60 muestra simplemente los mensajes que se irán mostrando mientras se procesa la información, con el fin de que los participantes no crean erróneamente que la aplicación ha dejado de funcionar.

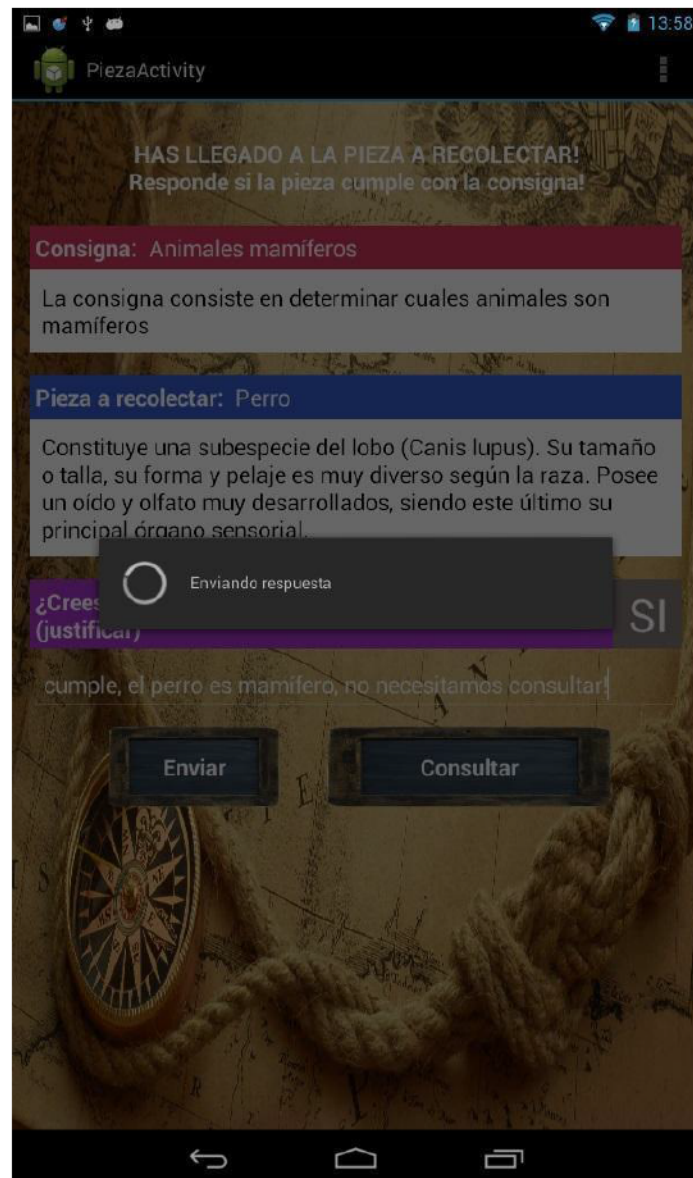


Figura 60. Mensaje de espera.

Como se puede apreciar en la Figura 61, al tomar una decisión sobre una de las piezas, esta cambia de color para que los participantes no se confundan y vuelvan a visitarla, de todos modos la aplicación ya desactivó el sensor de proximidad de esa pieza, por lo que si vuelven a visitarla no tendrá ningún efecto.

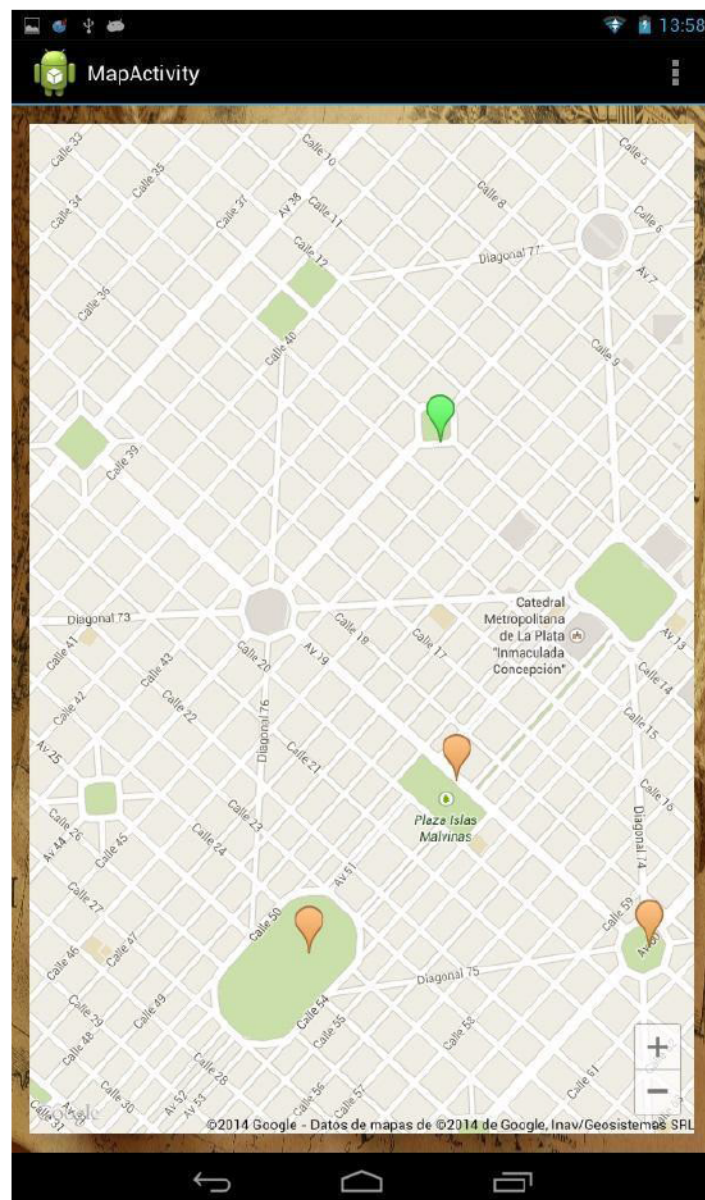


Figura 61. Una pieza visitada.

Supongamos que ahora el subgrupo se acerca a la pieza que está en la calle 12 entre 51 y 53, en las cercanías de la plaza Islas Malvinas.

Una vez situados en el lugar indicado, nuevamente aparecerá la información sobre una pieza a recolectar, en este caso “Surubi” (ver Figura 62). El proceder es similar al anterior, solo que en este caso supongamos que el subgrupo tiene dudas sobre la pieza y decide consultar. Entonces completa los campos correspondientes y pulsa el botón “Consultar”.

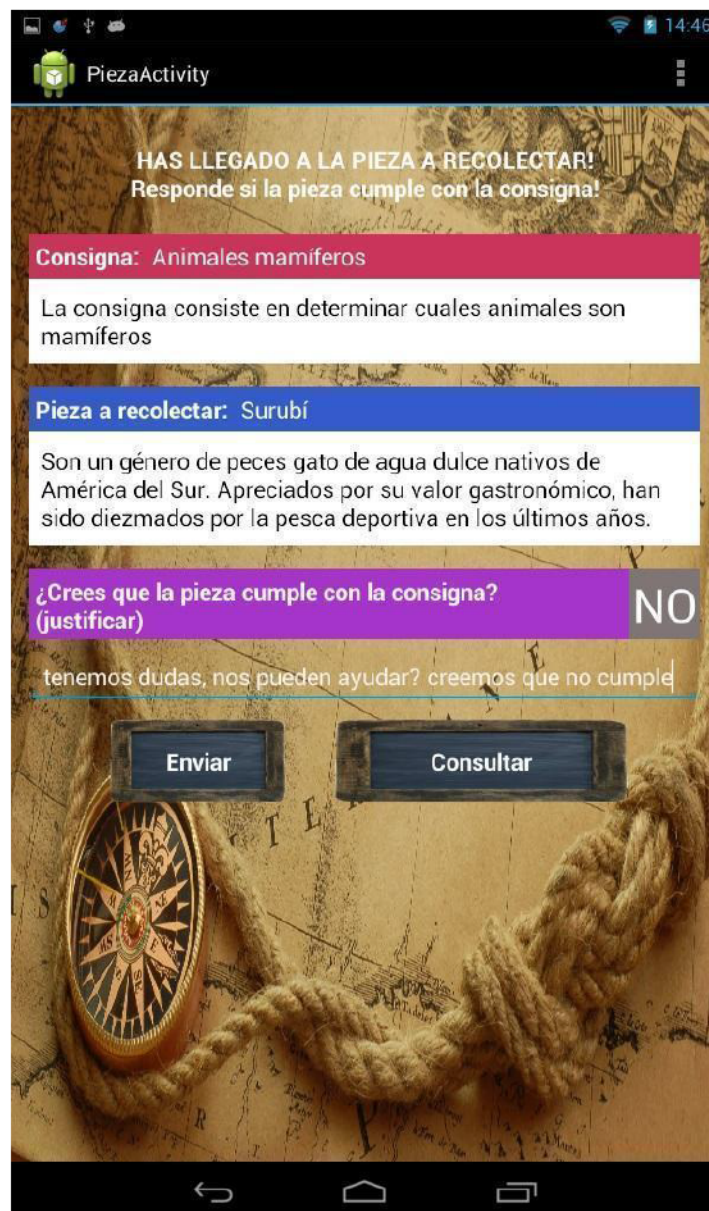


Figura 62. Pedido de ayuda.

Nuevamente el sistema les avisa a los participantes que está enviando la consulta a través de un mensaje en el dispositivo, como se puede visualizar en la Figura 63.

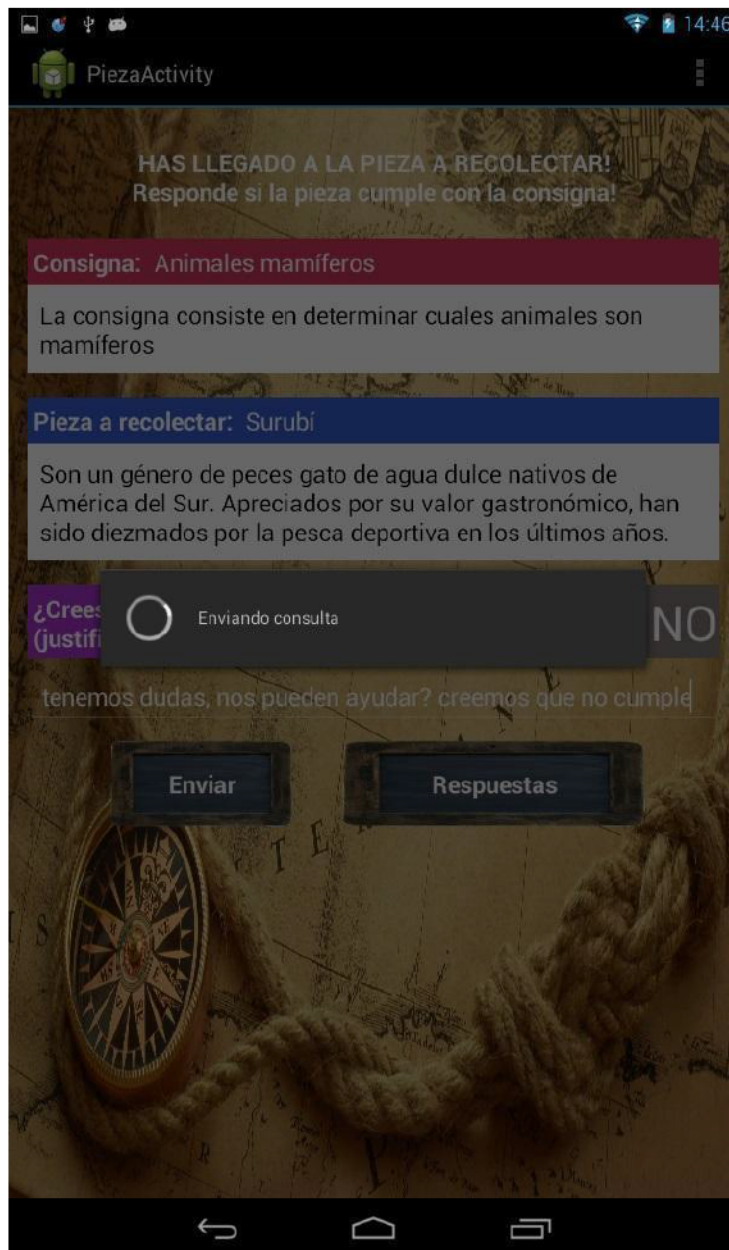


Figura 63. Mensaje de espera

En la Figura 64 se puede apreciar la pantalla donde los subgrupos ven las respuestas que les irán dando sus compañeros. Aquí pueden optar por esperar o bien, pulsar el botón “Tomar Decisión”, que los llevará nuevamente a la pantalla anterior y podrán tomar una decisión sobre la pieza sin esperar a que todos respondan.



Figura 64. Pantalla de Respuestas.

En la Figura 65, observamos como finalmente el Subgrupo 2 dio una respuesta y eso ayuda al Subgrupo 1 a responder la pregunta.

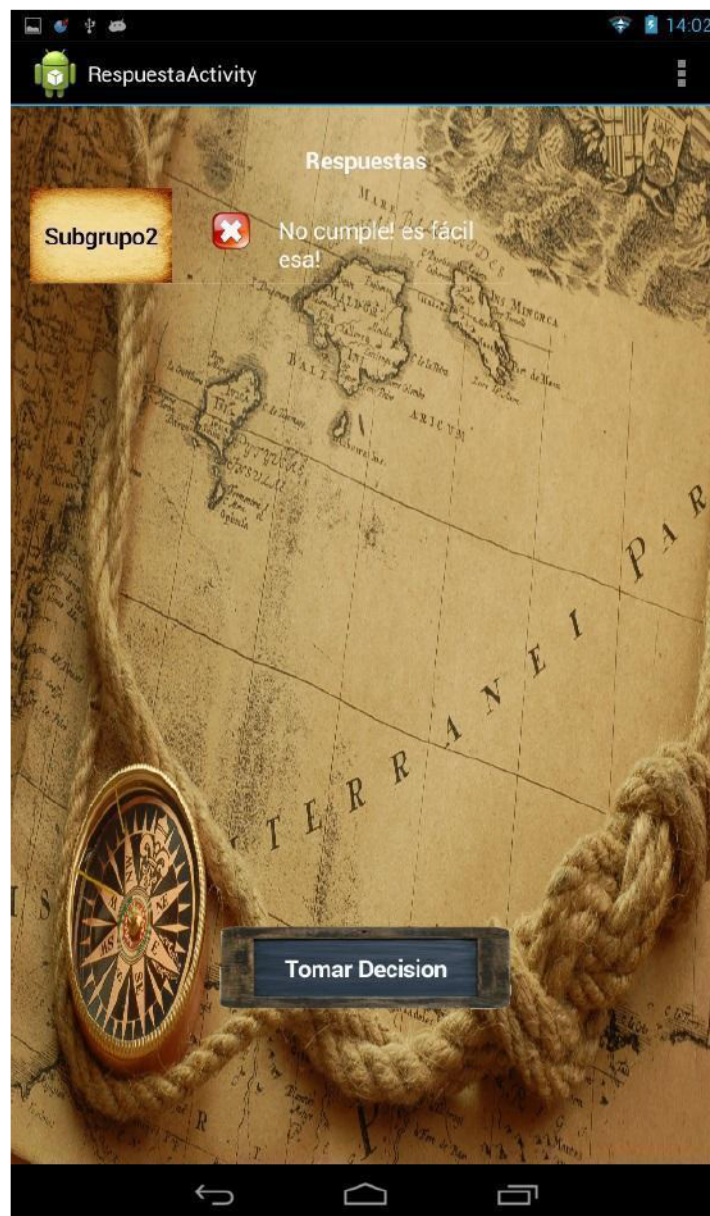


Figura 65. Respuesta del Subgrupo 2

Ayudado por los compañeros, el subgrupo puede responder correctamente la pregunta (ver Figura 66). De esta manera queda mostrado el modo de funcionamiento colaborativo.

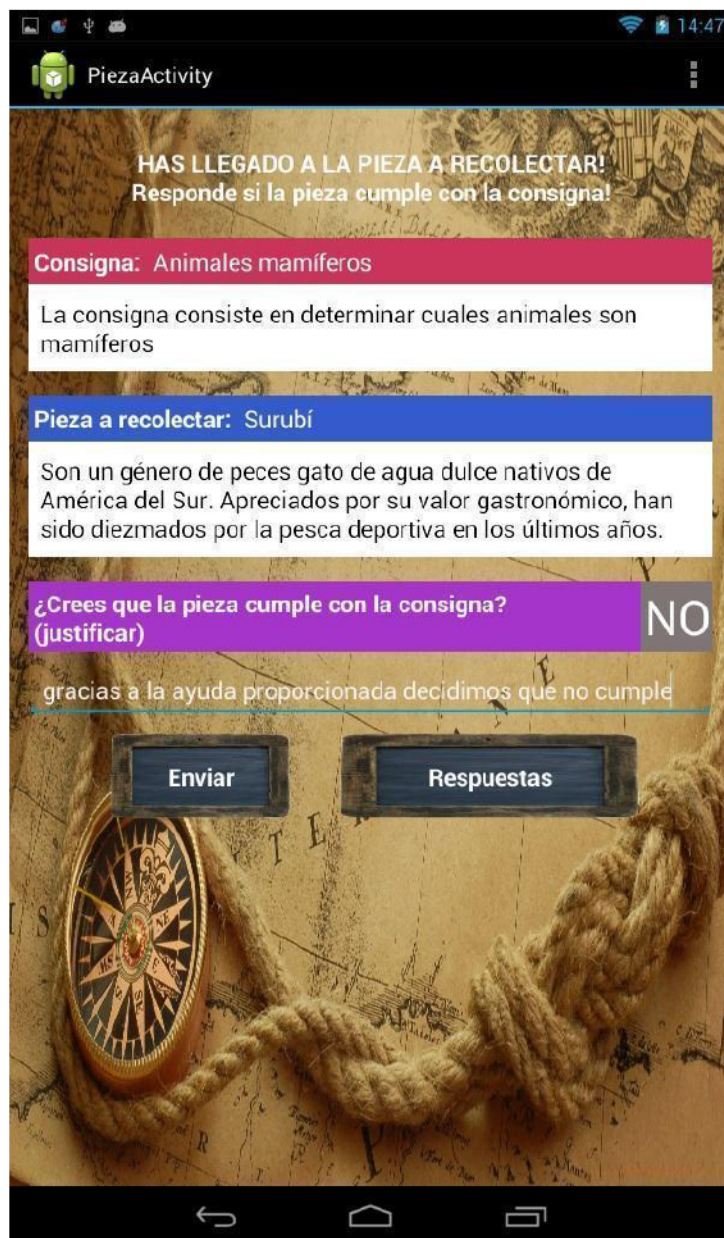


Figura 66 Decisión sobre la segunda pieza.

Una vez tomada la decisión sobre la pieza, el mapa se desplegará nuevamente actualizado (ver Figura 67), supongamos que ahora el subgrupo se dirige a la pieza a recolectar ubicada en el centro de la plaza Yrigoyen, en las intersecciones de las avenidas 60 y 19.

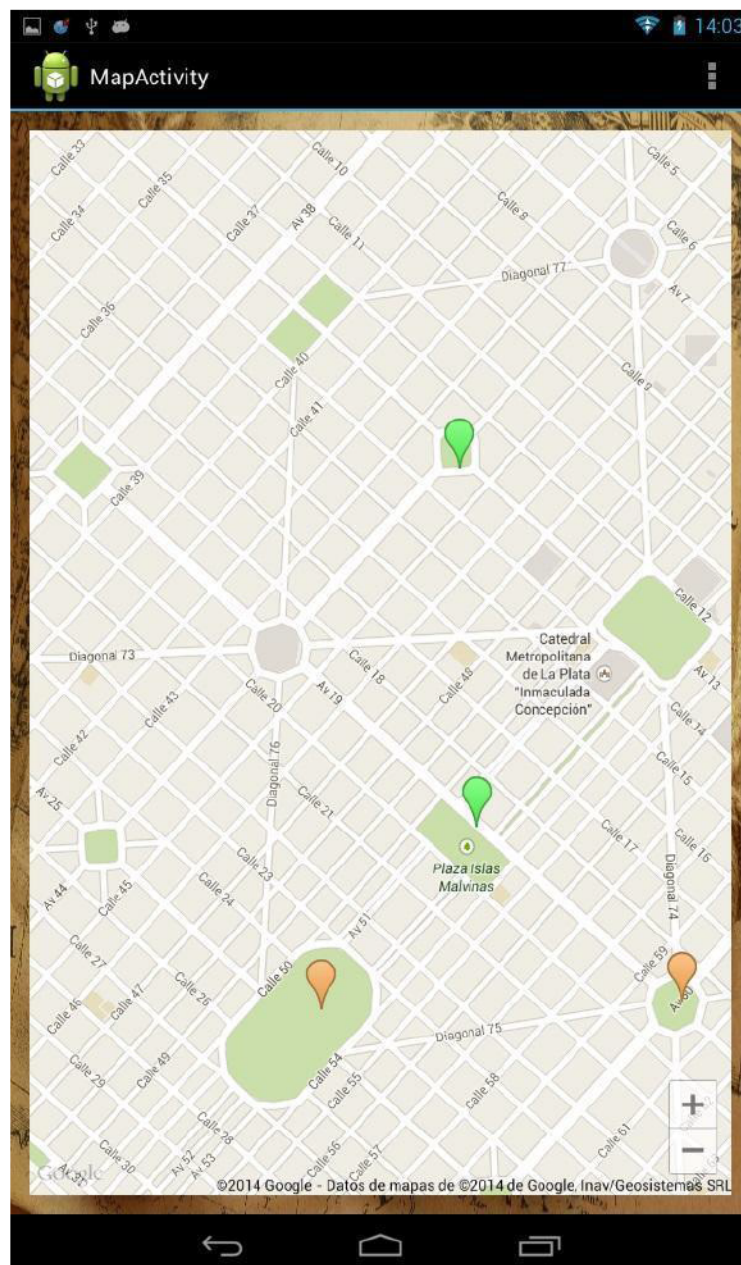


Figura 67. Mapa actualizado.

Mientras el Subgrupo 1 se dirigía a la plaza Yrigoyen, en el dispositivo desapareció el mapa y apareció un pedido de ayuda (ver Figura 68), otro subgrupo del grupo envió una consulta y se les debe responder.

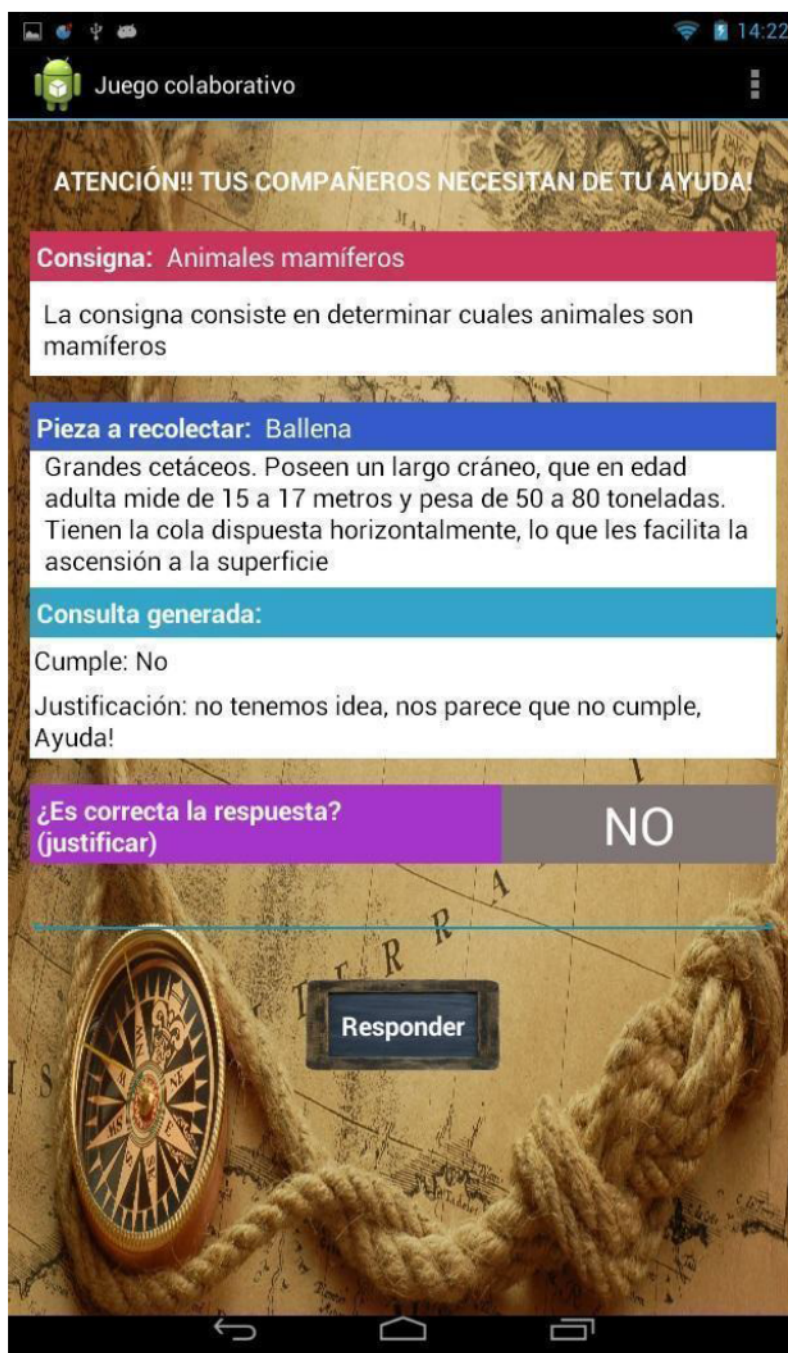


Figura 68. Pedido de ayuda.

Vemos en la Figura 68 como la consigna es la misma que la nuestra, luego se muestra la pieza a recolectar, "Ballena", con su descripción, e inmediatamente después se despliega la información sobre la consulta que generó el subgrupo, ellos piensan que no la ballena no es mamífero pero justifican diciendo que en realidad no lo saben, que necesitan ayuda. Lo que ahora hará el Subgrupo 1 es ayudar con lo que crea saber sobre la pieza, completando los campos correspondientes.

Luego de enviar la respuesta, en el dispositivo aparecerá un mensaje avisando que ya se envió la respuesta (ver Figura 69) y el Subgrupo 1 podrá continuar avanzando hacia la pieza a recolectar.

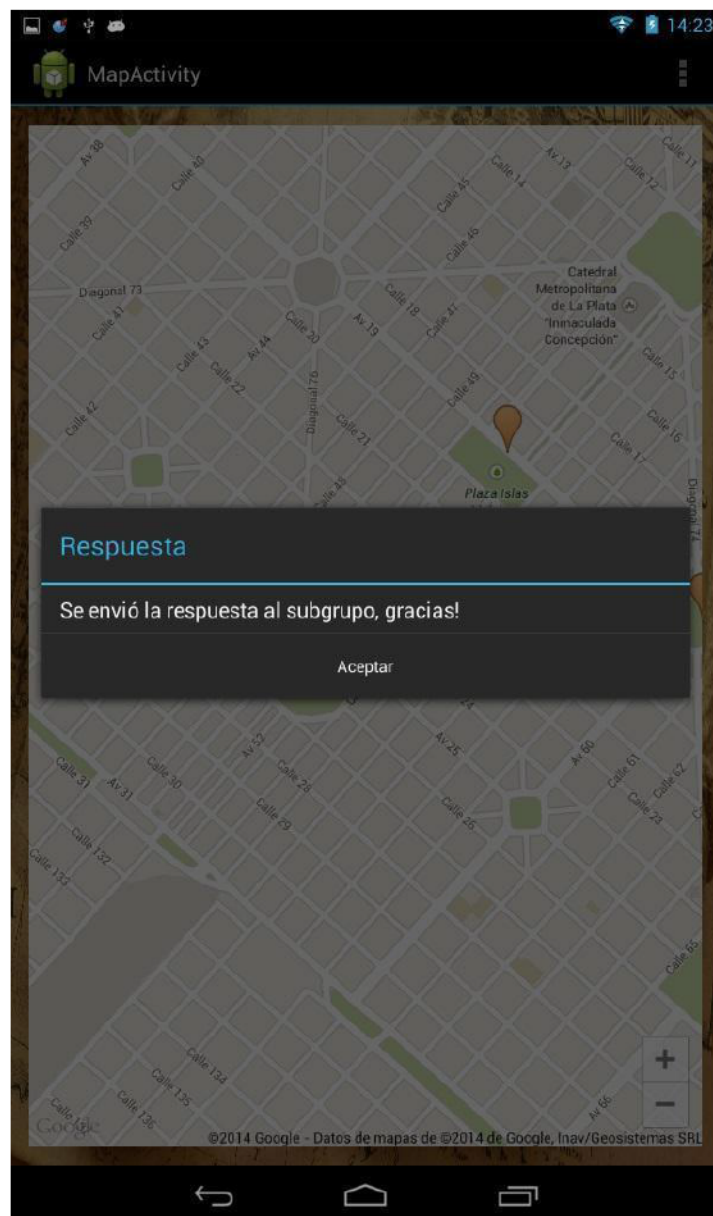


Figura 69. Mensaje de sistema agradeciendo ayuda.

Una vez enviada la respuesta, se despliega nuevamente el mapa (ver Figura 70) y el subgrupo sigue avanzando hacia 13 y 60. Vemos como se distingue claramente entre las piezas visitadas y las no visitadas.

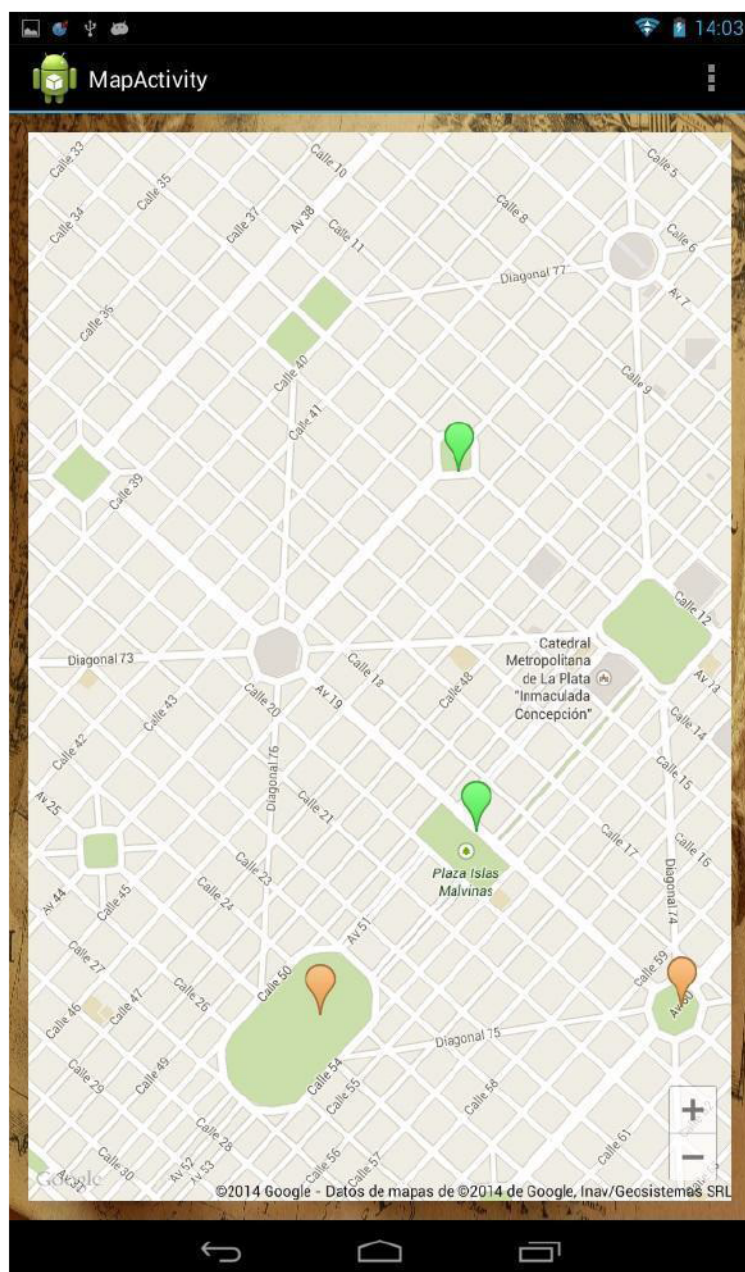


Figura 70. Mapa actualizado después de contestar la consulta.

Finalmente, el subgrupo arriba a la plaza donde está la tercer pieza a recolectar y se despliega la información necesaria, en este caso la pieza es el “Cisne”, el subgrupo decide no consultar y responde (erróneamente) y justifica su decisión final, como se muestra en la Figura 71.

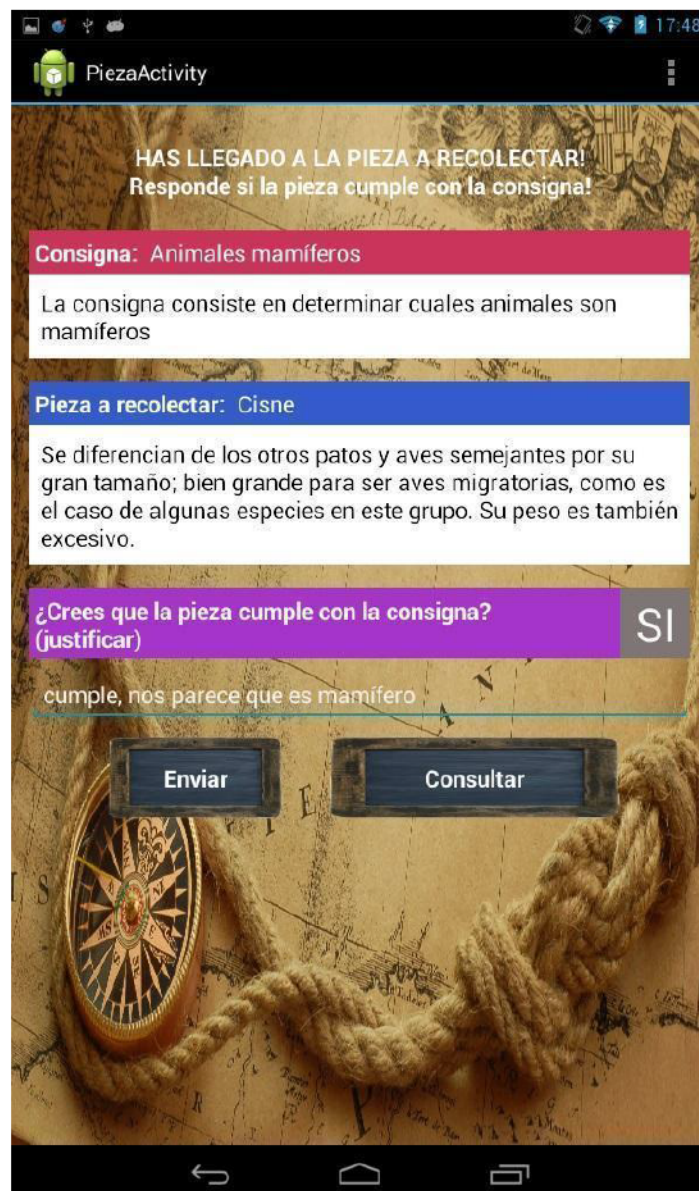


Figura 71. Información de la tercer pieza.

A modo de ejemplo vemos en la Figura 72 como la aplicación está preparada para soportar las vistas verticales y horizontales. Esto dependerá simplemente del dispositivo utilizado y de la comodidad de los participantes involucrados.

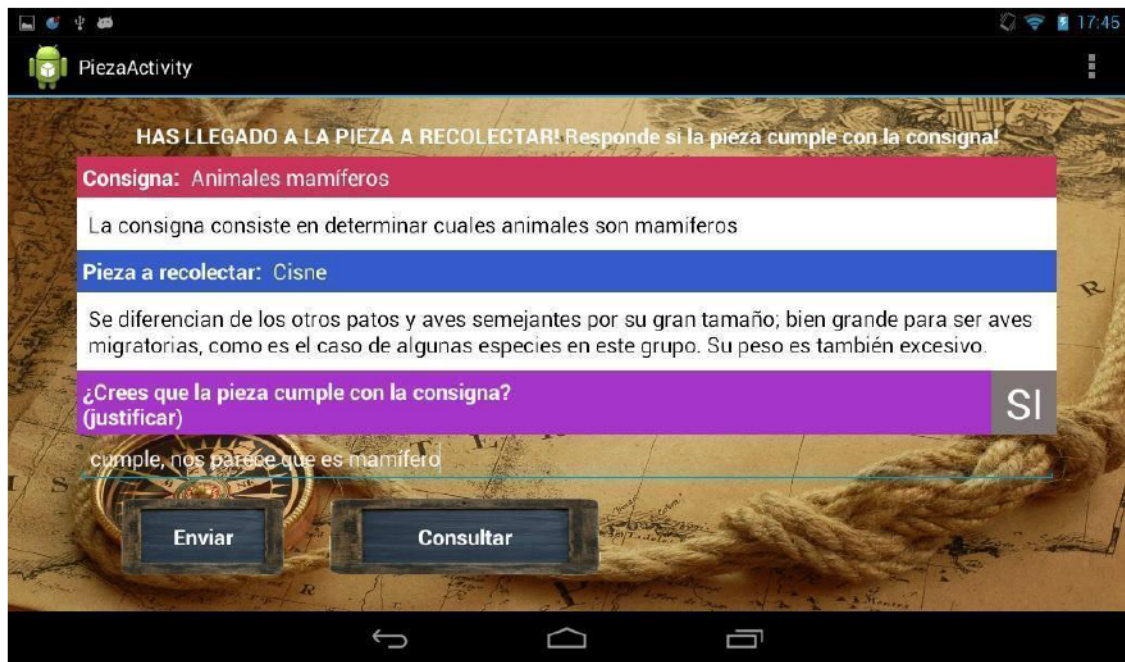


Figura 72. Información de la tercer pieza apaisada.

Observamos en la Figura 73 cómo el mapa se actualiza y queda solo una pieza por visitar, la que está ubicada en el parque Saavedra, a la altura de las calles 24 y 53. El subgrupo se dispone a llegar hasta allí.

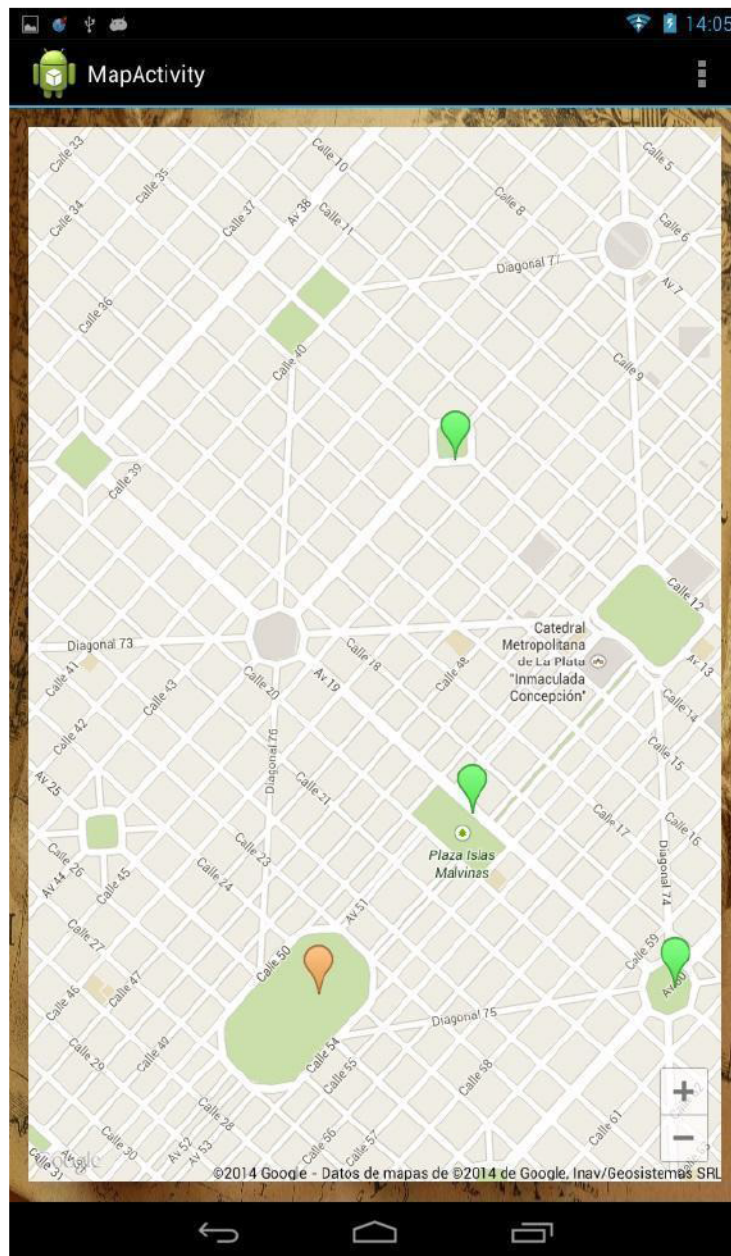


Figura 73. Mapa actualizado.

La última pieza a recolectar es “Murciélago”, el subgrupo tiene dudas sobre la decisión y decide consultar. En la Figura 74 se puede ver como luego de pulsar alguno de los botones, si faltan completar campos, se le mostrará un mensaje de advertencia y no los dejará continuar.



Figura 74. Mensaje de error.

Una vez completada la justificación, se puede enviar la consulta, como se muestra en la Figura 75.



Figura 75. Última pieza a recolectar

El subgrupo entonces completa el campo para justificar su decisión parcial y decide consultar al resto de los subgrupos.

Aquí haremos una salvedad, para ejemplificar mejor el caso de las respuestas de los demás subgrupos y la colaboración, en esta última pieza, haremos de cuenta que el grupo al cual pertenece el Subgrupo 1 posee cuatro subgrupos, de modo que al realizar una consulta, pueda esperar por tres respuestas (ver Figura 76).

Como vemos entonces, ahora el subgrupo puede recibir más ayuda, y está se irá desplegando automáticamente a medida que los subgrupos respondan. En la Figura 76 respondió el Subgrupo 4 indicándonos que el murciélago efectivamente es mamífero y aclarándonos que es uno de los pocos que vuelan.



Figura 76. Esperando respuestas.

En la Figura 77 se puede ver que respondió el Subgrupo 3, diciendo que creen que no pero no están seguros.



Figura 77 Llegada de una nueva respuesta.

Supongamos que el Subgrupo 1 decide responder sin esperar la respuesta del Subgrupo 2 (ver Figura 78).

Se toma la decisión de responder acorde a la ayuda recibida de los distintos compañeros de grupo.

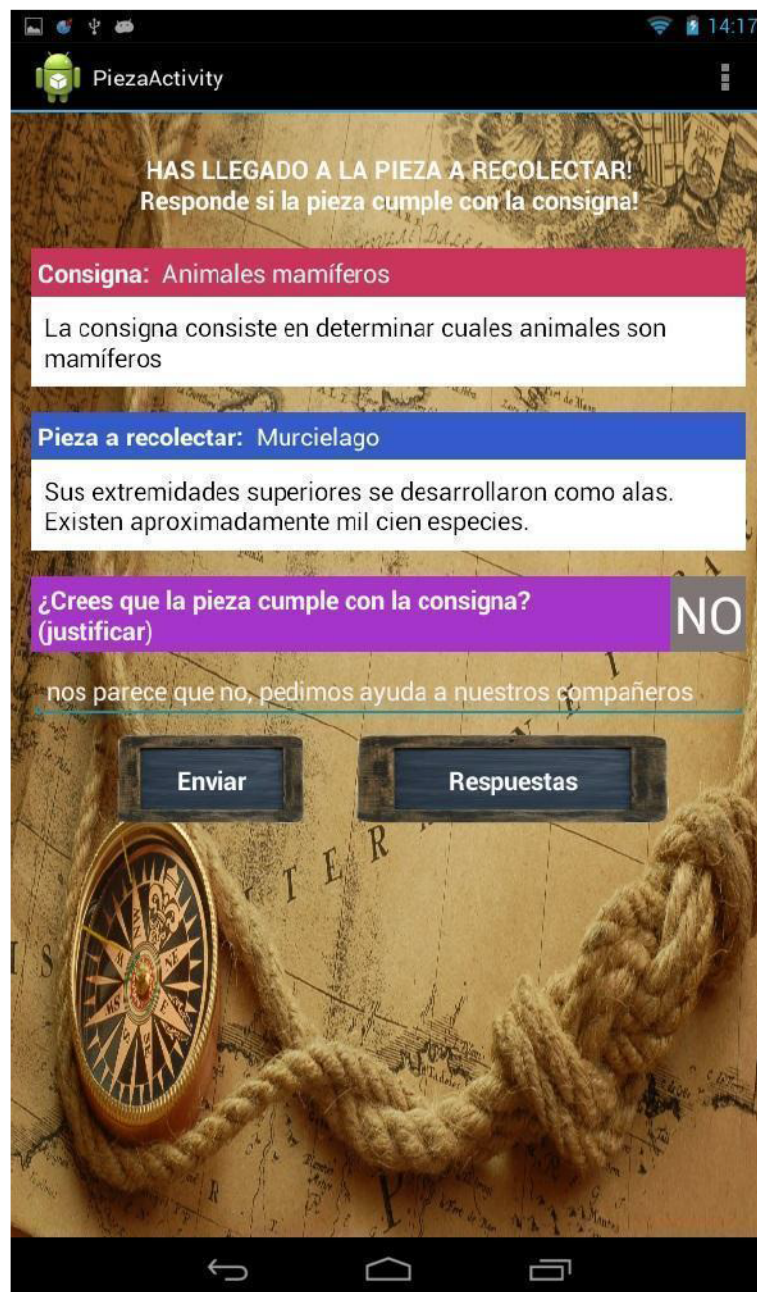


Figura 78. Última decisión.

Una vez resuelta la última pieza que se habían asignado, aparece un mensaje avisando que se terminó el recorrido como se puede observar en la Figura 79.

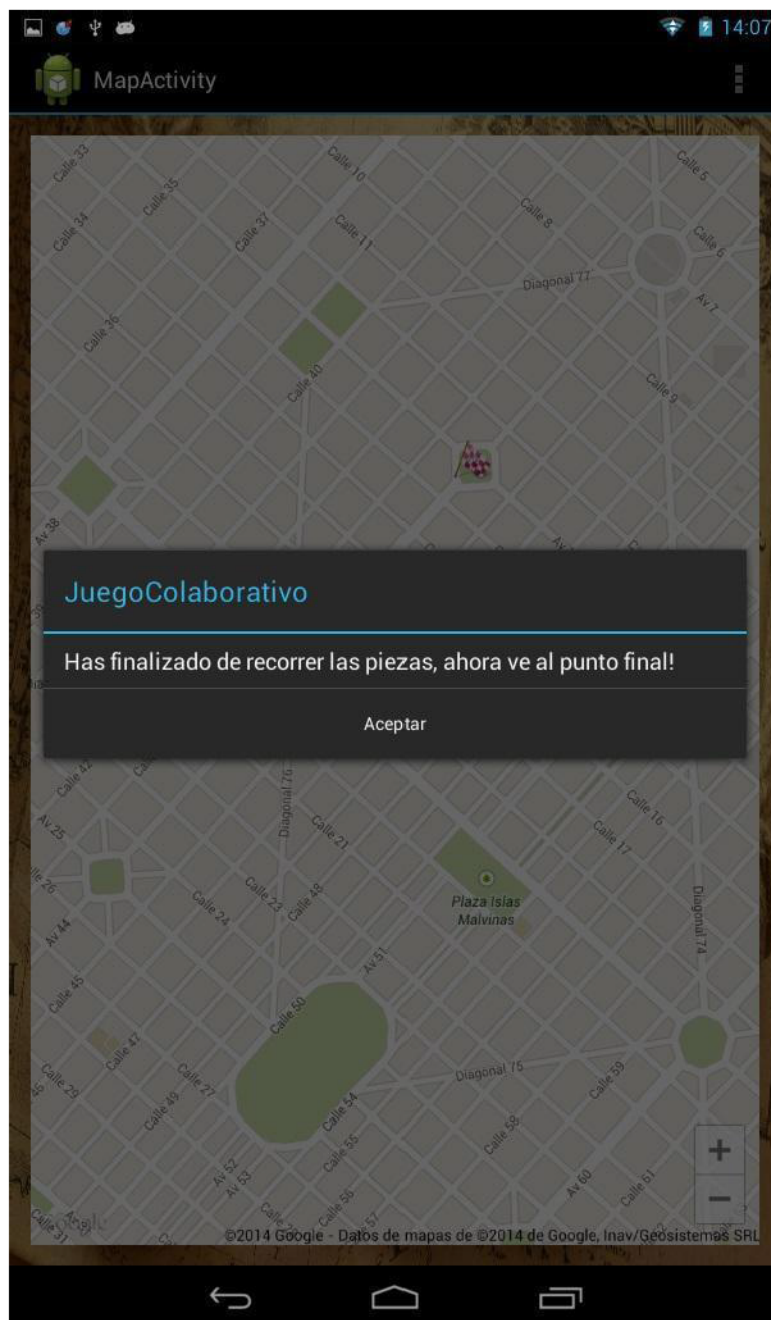


Figura 79. Comienzo del fin.

Luego los subgrupos deben dirigirse al punto de encuentro final del juego indicado en el mapa que reciben. Esto se puede apreciar en la Figura 80. Como se puede observar, en el mapa desaparecen las piezas recolectadas y aparece una bandera, en este caso roja, indicándonos el punto de encuentro final. En este caso ubicado nuevamente en plaza Paso, pero en otro de los vértices de la plaza.

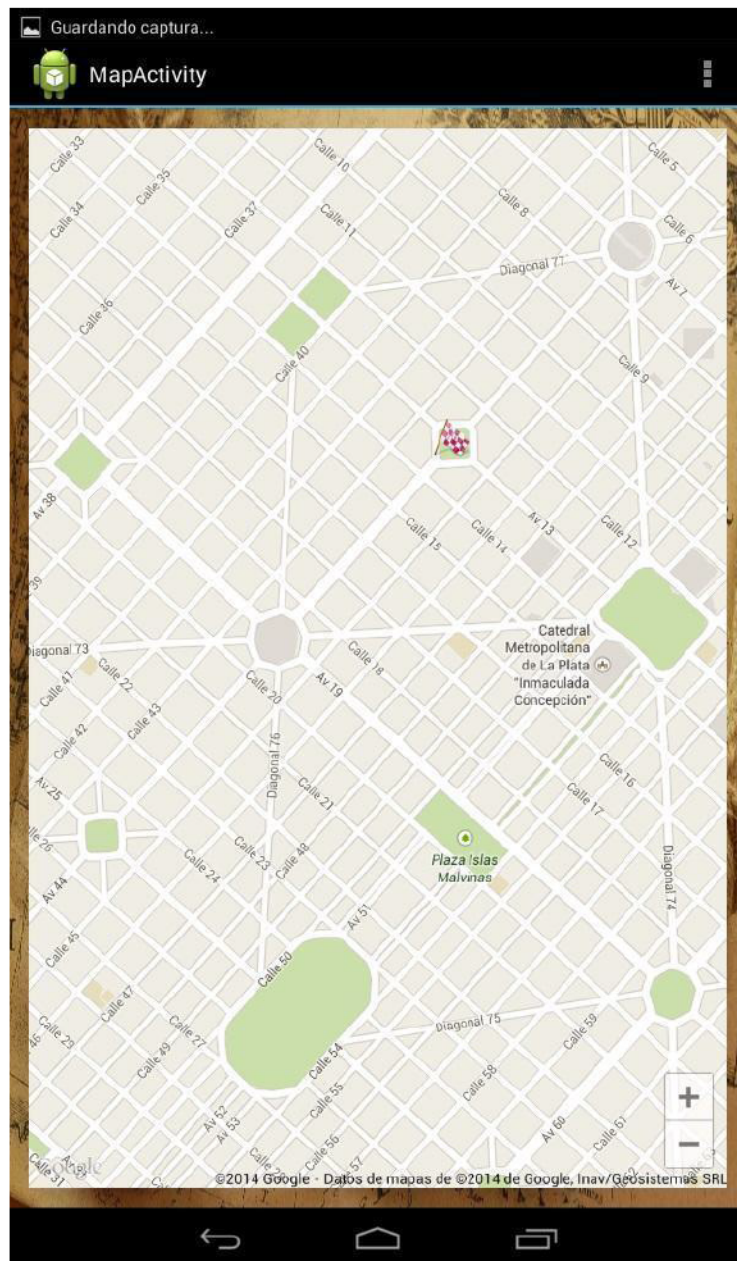


Figura 80. Bandera del punto de encuentro final.

Cuando el subgrupo arriba al punto de encuentro final, se despliega un mensaje advirtiendo la llegada y que se debe esperar la llegada de los demás compañeros de grupo (ver Figura 81). Del mismo modo que al comienzo del juego.

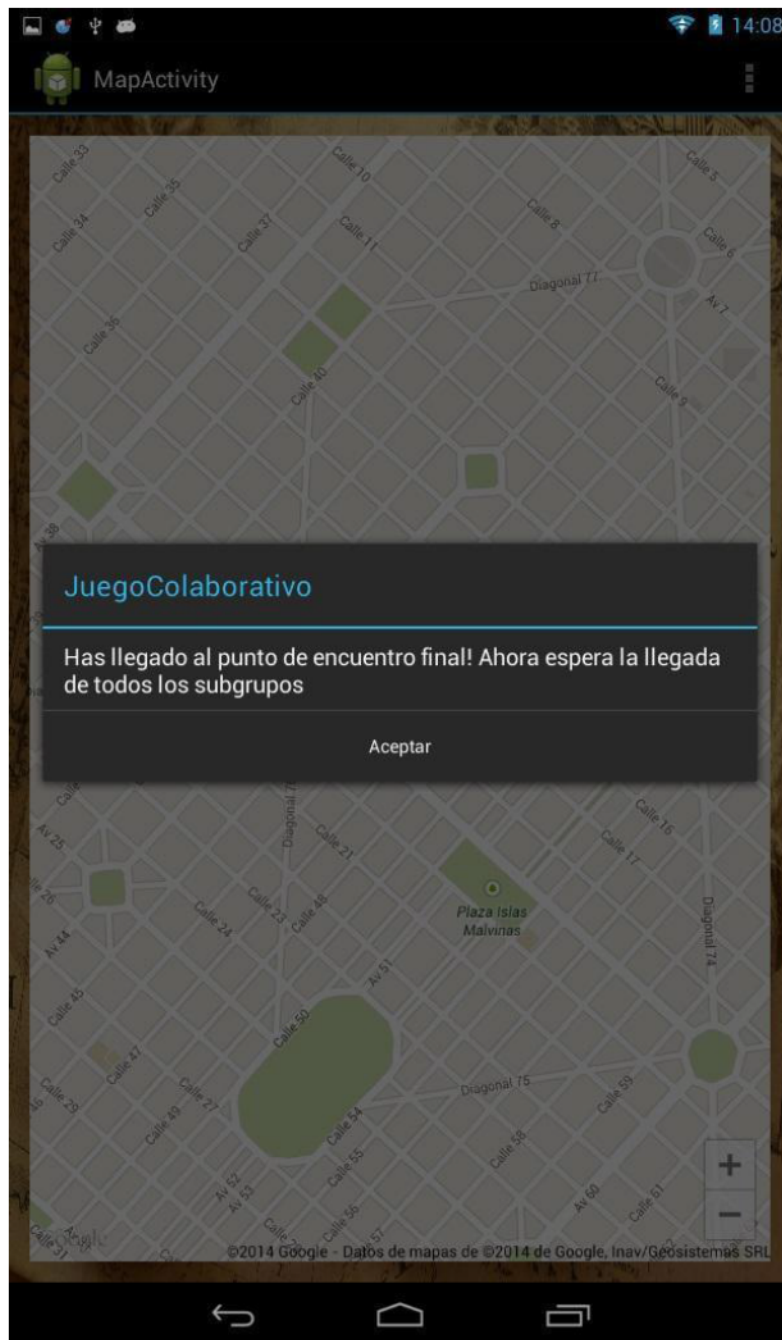


Figura 81. Llegando al punto de encuentro final.

Finalmente, una vez que llega el último subgrupo al punto final, el sistema calcula los puntajes y despliega en cada dispositivo los resultados generales a nivel grupal, y luego el detalle de la participación del subgrupo asociado al dispositivo.

Como se puede apreciar en la Figura 82 vemos como en esta simulación, se ha respondido las dos primeras piezas a recolectar de forma correcta y las dos restantes erróneamente.



Figura 82. Pantalla de resultados.

7. Conclusiones y Trabajos Futuros

7.1 Conclusiones

A partir de la investigación y el análisis en los temas relacionados con la computación móvil aplicada a la educación, desarrollamos un prototipo funcional estable de un juego colaborativo móvil. Se propuso un modelo que represente los principales conceptos involucrados en este tipo de juegos, el mismo se especificó de manera que pueda evolucionar y contemplar otras características como parte de estos juegos.

Nos propusimos facilitar una herramienta móvil que sirva de soporte para un aprendizaje colaborativo fuera del aula, que sea fácilmente configurable, dinámica y sencilla de comprender para los participantes involucrados.

Si bien es un prototipo, con sus limitaciones, hemos intentado que se asemeje lo más posible al modelo propuesto. Proveyendo la funcionalidad más compleja y relevante, dejando de lado implementaciones que no hacían al propósito general.

El aprendizaje colaborativo cumple un rol fundamental en el desarrollo de esta tesina. Teniendo en cuenta lo dicho, pusimos énfasis en este punto en todos los aspectos estudiados, tales como el marco teórico, el modelo del juego y el prototipo implementado.

Hemos logrado aunar diferentes tecnologías modernas, algunas con las cuales habíamos trabajado y otras con las que no. Esto nos representó un gran desafío en varios aspectos, en primer lugar, familiarizarnos con las nuevas tecnologías no conocidas, luego ponerlas en práctica; que funcionen correctamente y por último lograr que se acoplen sin inconvenientes. Tengamos en cuenta que, como vimos, tenemos un cliente móvil nativo en cada dispositivo, un servidor web y un servidor de base de datos.

Pusimos en práctica conocimientos adquiridos a lo largo de nuestro desarrollo universitario y adquirimos nuevos, que nos serán de gran ayuda en nuestro desenvolvimiento profesional.

Como resultado, además de realizar un prototipo funcional, a partir de la investigación y el análisis, hemos logrado introducirnos en aspectos relativos al aprendizaje colaborativo y móvil, incorporando conocimientos que nos permitirán involucrarnos en futuros desarrollos en temáticas vinculadas.

7.2 Trabajos Futuros

A continuación detallaremos lo que a nuestro entender se puede tomar como punto de partida a la hora de continuar o extender esta tesina de grado.

- Si bien en el capítulo cuatro se mencionó la capacidad de un subgrupo de tener asociado un dispositivo que le permita contemplar las características del mismo, en el prototipo funcional sólo se modela esta característica, sin la correspondiente implementación. Asumimos que los subgrupos cuentan con un dispositivo con las características mínimas para poder ejecutar el juego y desarrollarlo sin mayores inconvenientes (GPS, Google Play --necesario para el google map--, paquete de datos).
- Un aspecto del juego que nos pareció que se puede mejorar en cuanto a eficiencia es el sistema de consultas constantes al servidor para determinar algún evento, por ejemplo, cuando se instancia un servicio que envía una solicitud cada un intervalo de tiempo determinado para consultar al servidor si todos los subgrupos están en un estado en particular. Por cuestiones de alcance de la tesina y tiempo de implementación decidimos utilizar esta forma de comunicación para el desarrollo del prototipo. Existen algunos métodos alternativos de comunicación (como por ejemplo Google Cloud Message para Android [56]) en el que el servidor le notifica a todos los clientes un evento en particular en lugar de que los clientes estén preguntando constantemente por este evento.
- El cliente no tiene recuperación ante caídas o fallos del sistema. Si ocurre un error y la aplicación se detiene o el dispositivo se apaga, hay que volver a iniciar el juego. Y el subgrupo que inicia nuevamente, inicia sobre la posición inicial del juego, como si comenzase nuevamente. Falta simplemente la recuperación del estado e información de un subgrupo al momento del reinicio, dado que esta información está almacenada en el servidor. Se podría enviar dicha información al cliente cuando el subgrupo se loguea en la aplicación y así, volver al último estado “estable” del subgrupo luego del fallo.
- Ante un fallo en el prototipo, en algunos casos el sistema se detiene, sin mostrar la información, o no se detiene pero muestra información del error muy acotada, simplemente avisa cual tarea del cliente que estaba intentando procesar falló. No detalla los motivos, por ejemplo si el login falla porque no se puede conectar con el servidor, en pantalla solo muestra un mensaje con la descripción “fallo en la tarea login”. La aplicación debería ser capaz

informar un detalle del error más específico y generar el log correspondiente para un futuro análisis y poder evitar estos errores en futuras versiones.

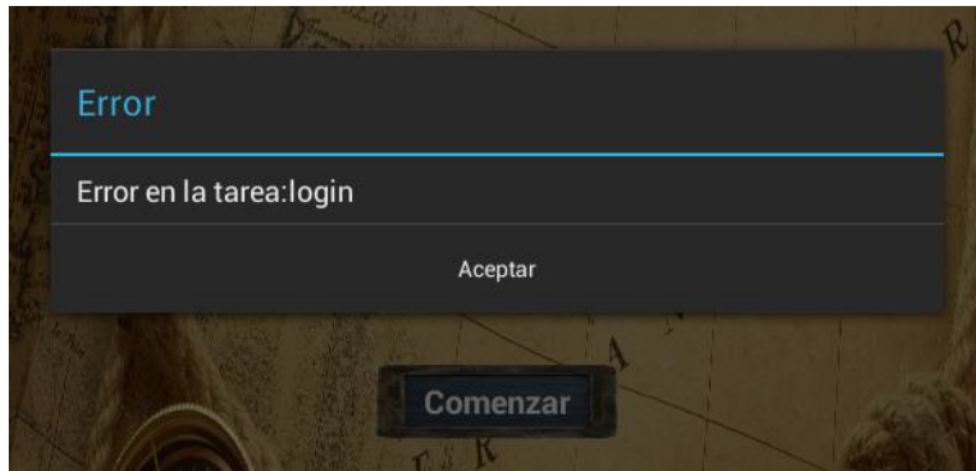


Figura 83. Error en el login, se muestra información sobre la tarea que falló, en este caso fue por no poder establecer conexión con el servidor.

- La aplicación es pobre en cuanto a la transmisión de excepciones del lado del servidor hacia las aplicaciones nativas. En el servidor se conocen las excepciones que ocurren durante el transcurso del juego, sin embargo, no son transmitidas a las aplicaciones nativas de forma adecuada. Podrían buscarse alternativas a la hora de transmitir las excepciones, como por ejemplo, manejarlas mediante soap faults [56] y poder recibirlas de forma correcta en las aplicaciones nativas utilizando las librerías disponibles en Android para manejar comunicación mediante SOAP.
- Como dijimos, nosotros resolvimos las llegadas de los grupos a los POIS de forma automática utilizando sensores de proximidad, pero no explotamos otras alternativas y ni probamos que funcione en ambientes reales. Podría estudiarse en profundidad esta sección y evaluarse otros métodos, como por ejemplo utilizando lecturas de códigos QR [55].
- Hay algunos aspectos específicos para mejorar del prototipo que debemos mencionar:
 - Agregar la posibilidad de tomar fotografías del lugar o generar videos para fomentar y auxiliar la colaboración entre los participantes.
 - Alertas para encender WIFI y GPS en los dispositivos al iniciar la aplicación.
 - Generar vibraciones en los dispositivos ante la aparición de nuevos eventos.
 - En la administración web, agregar en la pantalla donde se visualizan todas las piezas, la opción de resaltar aquellas que cumplen con determinada consigna.

- Ampliar la información de las piezas a recolectar, por ejemplo, con imágenes de la misma.
 - Dado que no era el objetivo principal de la tesis y no se le dió prioridad, se podría mejorar el aspecto visual, tanto de la parte web, como del cliente Android.
- Si bien es un prototipo y no estaba en el alcance de esta tesina, sería deseable poder comenzar a realizar trabajos de campo que permitirán afianzar, modificar corregir y ampliar el desarrollo realizado.

8. Bibliografía

01. Dirección de Investigación y Desarrollo Educativo Vicerrectoría Académica, Instituto Tecnológico y de Estudios Superiores de Monterrey. Las estrategias y técnicas didácticas en el rediseño: Aprendizaje Colaborativo. 2001.
<http://sitios.itesm.mx/va/dide2/tecnicas_didacticas/ac/Colaborativo.pdf>
02. Pivec, Paul. Aprendizaje basado en juegos: nuevas prácticas, nuevas aulas. E-Learning Papers. 2011.
03. Echols, Michael. Going social. Chief Learning Officer. 2011.
04. De Freitas, S., Oliver, M. How can exploratory learning with games and simulations within the curriculum be most effectively evaluated? School of Computer Science and Information Systems, Birkbeck College, University of London. 2006.
05. Kohnstamm, Jacob. Opinion on Geolocation services on smart mobile devices. ARTICLE 29 Data Protection Working Party. 2011.
<http://ec.europa.eu/justice/policies/privacy/docs/wpdocs/2011/wp185_en.pdf>
06. Lyytinen, K. and Yoo, Y. Issues and challenges in ubiquitous computing. 2002.
07. Roy, N., Scheepers, H. and Kendall, E.: Mapping the Road for Mobile Systems Development. In Proceedings of Pacific Asia Conference on Information Systems, paper 94, pp. 1358-1371. 2003.
08. Talukder, A.K. and Yavagal, R. Mobile Computing: Technology, Applications, and Service Creation. McGraw-Hill Professional. 2006.
09. Talukder, A.K., Ahmed, H. and Yavagal, R. Mobile Computing: Technology, Applications, and Service Creation. Second Edition. McGraw-Hill Professional. 2010.
10. Zambrano Polo, La Borda. Integrando Sensibilidad al Contexto Mediante Aspect Oriented Programming. 2006.
11. Weiser, Mark. Some computer science problems in ubiquitous computing. Communications of the ACM, July:137–143, 1993.
12. Hansen Frank Allan, Olof Bouvin Niels. Mobile Learning in Context – Context-aware Hypermedia in the Wild. 2009
13. Hendrik Thüs, Mohamed Amine Chatti, Esra Yalcin, Christoph Pallasch, Bogdan Kyrlyiuk, Togrul Mageramov, and Ulrik Schroeder. Mobile Learning in Context. 2012.
14. Anind K. Dey and Gregory D. Abowd. Towards a Better Understanding of Context and Context-Awareness. 1999.
15. Juan Felix Basterretche. Dispositivos Móviles. L.U.: 34039. 2007
16. J. Huizenga, W. Admiraal, S. Akkerman and G. ten Dam. Mobile game-based learning in secondary education: Engagement, motivation and learning in a mobile city game. 2009.

17. Joost Raessens. Playing History. Reflections on Mobile and Location-Based Learning. 2007
18. Angel Riviére. Teoría social aprendizaje. Implicaciones Educativas. 1992
19. Frank Allan Hansen, Karen Johanne Kortbek, Kaj Grønbaek. Mobile Urban Drama for Multimedia-Based Out-of-School Learning. 2010.
20. Crompton Helen. A Historical Overview of M-Learning. 2013.
21. Carmen Cantillo Valero, Margarita Roura Redondo, Ana Sánchez Palacín. Tendencias actuales en el uso de dispositivos móviles en educación. Junio 2012
22. Darrell M. West. Mobile Learning: Transforming Education, Engaging Students, and Improving Outcomes. 2013.
23. Stefan Kalms, Dorothee Zerwas, Harald F. O. Von Kortzfleisch. Ubiquitous Entrepreneurship. 2013
24. Gwo-Jen Hwang, Tzu-Chi Yang, Chin-Chung Tsai, Stephen J.H. Yang. A context-aware ubiquitous learning environment for conducting complex science experiments. 2008
25. Sara De Freitas. Learning in Immersive worlds A review of game-based learning. 2006.
26. Scagnoli Norma et al. Aprendizaje Colaborativo. 2005.
27. Jordi Miralles. Juegos cooperativos, un compromiso social ineludible. Fundación Tierra. 2006.
28. Steve Benford, Duncan Rowland, Martin Flintham, Adam Drozd, Richard Hull, Josephine Reid, Jo Morrison, Keri Facer. Life on the Edge Supporting Collaboration in Location-Based Experiences. 2005.
29. Guy Redwood. Mobile Gaming: A Usability Study. 2012.
<<http://www.simpleusability.com/beinspired/2012/01/mobile-gaming-usability>>
30. Sharples, M., Taylor, J., Vavoula, G.: A theory of learning for the mobile age. 2007
31. Krasner, G. E., & Pope, S. T. A description of the model-view-controller user interface paradigm in the smalltalk-80 system. Journal of object oriented programming, 1(3), 26-49. 1988.
32. <http://www.apache.org/>
33. <http://www.mysql.com/>
34. <http://www.php.net/>
35. <http://www.android.com/>
36. <http://symfony.com/>
37. http://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos
38. Charisma Template <<http://usman.it/free-responsive-admin-template/>>
39. Android SDK <<https://developer.android.com/sdk/index.html>>
40. <http://developer.android.com/intl/es/google/play-services/maps.html>
41. <http://jquery.com/>
42. <http://gmap3.net/>
43. <https://developers.google.com/maps/documentation/android/marker>
44. <https://www.java.com/>

45. <http://www.symfony-project.org/plugins/ckWebServicePlugin>
46. <https://code.google.com/p/ksoap2-android/>
47. http://es.wikipedia.org/wiki/Simple_Object_Access_Protocol
48. <http://www.w3schools.com/Webservices/>
49. <http://developer.android.com/intl/es/reference/android/os/AsyncTask.html>
50. <http://developer.android.com/intl/es/guide/components/processes-and-threads.html>
51. <http://es.wikipedia.org/wiki/WSDL>
52. <https://play.google.com/store/apps/details?id=com.fakegps.mock>
53. <http://www.genymotion.com/>
54. <https://developer.android.com/intl/es/google/gcm/index.html>
55. http://es.wikipedia.org/wiki/C%C3%B3digo_QR
56. <http://developer.android.com/intl/es/google/gcm/index.html>
57. http://www.w3schools.com/webservices/ws_soap_fault.asp